In compliance with the
Canadian Privacy Legislation
some supporting forms
may have been removed from
this dissertation.

While these forms may be included
in the document page count,
their removal does not represent
any loss of content from the dissertation.

University of Alberta

CHARACTERIZATION OF HIGH ORDER CORRELATION FOR ENHANCED INDICATOR SIMULATION

by

Julián Maximiliano Ortiz Cabrera ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

in

Mining Engineering

Department of Civil and Environmental Engineering

Edmonton, Alberta
Fall 2003

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisisitons et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-88029-X
Our file Notre référence
ISBN: 0-612-88029-X

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

Canada

University of Alberta

Library Release Form

Name of Author: Julián Maximiliano Ortiz Cabrera

Title of Thesis: Characterization of High Order Correlation for Enhanced Indicator Simulation

Degree: Doctor of Philosophy

Year this Degree Granted: 2003

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Date: June 5, 2003

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Characterization of High Order Correlation for Enhanced Indicator Simulation** submitted by Julián Maximiliano Ortiz Cabrera in partial fulfillment of the requirements for the degree of **Doctor of Philosophy** in *Mining Engineering.*

Dr. C

Dr. Dwayne D. Tannant (Examiner)

Date: June 5, 2003

To Mabel, Amanda, Martín, and Vicente.
To my parents Lucía and Eduardo.

Abstract

Geostatistical simulation aims at reproducing the variability of the real underlying phenomena. When non linear features or large range connectivity are present, the traditional simulation approaches that use only two-point statistics, such as a variogram or covariance function, do not provide good reproduction of those features. Connectivity of high and low values is often critical for grades in a mineral deposit, concentrations of a pollutant in an environmental study, or high permeability flow paths in a petroleum reservoir. Multiple-point statistics can help to characterize these features.

The use of multiple-point statistics in geostatistical simulation was proposed more than ten years ago, based on the use of training images to extract the statistics. This research proposes the use of multiple-point statistics extracted from actual data.

A simulation method is developed to account for runs, that is, strings of points that are all above (or below) a threshold. The method is implemented in a hierarchical fashion, starting at the highest threshold and eroding the field to reproduce histograms of runs above and below several thresholds. A selection function is used to pick the nodes that will be switched to be below the threshold (eroded). Implementation shows that the selection function is critical to obtain convergence to the target histograms of runs. However, artifacts were found that invalidate this approach.

A second approach is proposed to correct the indicator kriging probabilities used in sequential indicator simulation, with probabilities extracted from multiple-point configurations. The correction is done under three different assumptions of redundancy between the two sources of information. The practical implementation of these methods showed improvement in the numerical models for medium and long term mine planning.

Acknowledgements

I would like to thank my advisor, Dr. Clayton V. Deutsch, for his help and support throughout this research. This study benefitted from his creativity and experience. His wide knowledge about the practice and theory of geostatistical methods has been an inspiration to push my curiosity beyond what I thought was its limit.

I am also grateful to my undergraduate ore reserve estimation professor back in Chile, Dr. Eduardo Magri, for giving me the motivation and guidance to start graduate studies.

I am most grateful to the Mining Department at the Universidad de Chile for his financial support, which would not have been possible without the help of his industry affiliates, in particular Codelco Chile. The Centre for Computational Geostatistics was another source of financial stability during these years and I am grateful to the industry affiliates that keep supporting the research developed by the group.

I thank all the students that I met during these years for their companionship. In particular, I thank Oy Leuangthong, Karl Norrena, and Michael Pyrcz for their friendship and for showing me the Canadian approach to life (and to winter).

Finally, I thank my wife Mabel for her support and love during these years, and my children, Amanda, Martín, and Vicente for bringing endless happiness to my life.

Table of Contents

1 **Introduction** 1

1.1 Problem Setting 1

1.2 Proposed Approach 4

1.2.1 Accounting for Multiple-Point Statistics as Runs 4

1.2.2 Integrating the Indicator Kriging Probability and the Multiple-Point Statistics under Different Assumptions 5

1.3 Dissertation Outline 5

2 **Overview of Geostatistics** 7

2.1 Theory of Regionalized Variables 8

2.2 Statistical Inference and Stationarity 9

2.2.1 Moments of a Random Variable 10

2.2.2 Decision of Stationarity 11

2.2.3 Inferring Representative Histograms 11

2.2.4 Variogram Inference 13

2.2.5 Inferring Multiple-Point Statistics 14

2.2.6 A Note on Positive Definiteness 15

2.3 Geostatistical Estimation 16

2.3.1 Simple Kriging 16

2.3.2 Ordinary Kriging 18

2.3.3 Non-Stationary Kriging 19

2.3.4 Non Linear Variants 19

2.3.5 Cokriging 20

2.3.6 Indicator-Based Estimation 21

2.4 Conventional Two-Point Geostatistical Simulation 30

2.4.1 The Place of Simulation 30

2.4.2 Gaussian Techniques 31
2.4.3 Indicator Simulation 36
2.5 Attempts at Multiple-Point Geostatistics 37
2.5.1 Object-Based Methods 38
2.5.2 Variogram-Based Techniques 38
2.5.3 N-Point Connectivity Function 38
2.5.4 Extended Normal Equations 39
2.5.5 Simulated Annealing 41
2.5.6 Iterative Methods and Markov Chain Monte Carlo Methods . 44
2.6 Discussion 45

3 Incorporating Multiple-Point Runs in Geostatistical Simulation 47
3.1 Introduction 47
3.1.1 Key Concepts 47
3.1.2 The Theory of Runs 50
3.2 Analytical Derivation of the Frequency of Runs 52
3.2.1 General Case 53
3.2.2 The Multi-Gaussian Case 53
3.2.3 Example 54
3.2.4 The Random Case: Relation with Mood's Results 55
3.2.5 Discussion 57
3.3 Hierarchical Indicator Simulation 57
3.3.1 Methodology 57
3.3.2 Implementation Problems 67
3.3.3 Examples 69
3.4 Comments on the Direct Simulation of Runs 86

4 Updating the Indicator Kriging Probability with Multiple-Point Statistics 89
4.1 Introduction 89
4.2 Statistical Inference of Two-Point and Multiple-Point Statistics . . . 91
4.3 Integrating Multiple-Point Statistics 91
4.3.1 Assumption of Independence Between Multiple Events 92
4.3.2 Permanence of Ratios Assumption 92
4.3.3 Multi-Gaussian Assumption 93

4.3.4 Comments 96
4.4 Practical Implementation 97
4.4.1 Sequential Multiple-Point Simulation 98
4.5 Applications 100
4.5.1 Binary Examples 100
4.5.2 Continuous Variable Example 107
4.5.3 Discussion 107
4.6 Assessing Performance 111
4.7 Quantifying Non-Convexity on the Estimators 115
4.8 Discussion 115

5 Case Study **119**

5.1 Introduction 119
5.2 Available Data and Basic Statistics 120
5.2.1 Drillhole Information 120
5.2.2 Blasthole Information 122
5.2.3 Declustering 125
5.2.4 Comparison of Datasets 125
5.2.5 Comments 129
5.3 Variogram Modelling 132
5.3.1 Selection of Thresholds 132
5.3.2 Variogram Calculation and Modelling 134
5.4 Multiple-Point Statistics Inference 136
5.5 Sequential Indicator Simulation 140
5.5.1 Parameters 140
5.5.2 Validation of Results 140
5.6 Assumption of Independence between Single-Point (DH Data) and Multiple-Point Information (BH Data) 148
5.6.1 Parameters 148
5.6.2 Validation of Results 148
5.7 Assumption of Permanence of Ratios 151
5.7.1 Parameters 151
5.7.2 Validation of Results 152
5.8 Multi-Gaussian Assumption 157

5.8.1 Parameters 157
5.8.2 Validation of Results 157
5.9 Sequential Gaussian Simulation 161
5.9.1 Normal Score Transformation 168
5.9.2 Variogram of Normal Scores 168
5.9.3 Parameters 168
5.9.4 Validation of Results 169
5.10 Comparison of Results 173
5.10.1 Statistical Performance 173
5.10.2 Mine Planning Performance 173
5.10.3 Conclusions 175

6 Conclusions and Future Work **179**
6.1 Conclusions 179
6.1.1 Incorporating Multiple-Point Runs in Geostatistical Simulation 179
6.1.2 Updating the Indicator Kriging Probability with Multiple-Point Statistics 180
6.2 Future Work 182

Bibliography **197**

A Pseudo-Random Number Generators **199**
A.1 Random Number Generators 199
A.1.1 Linear Congruential Method 200
A.1.2 Additive Congruential Method 200
A.1.3 Other Methods 201
A.2 Statistical Tests 201
A.2.1 Empirical Tests 202
A.2.2 Theoretical Tests 205
A.3 Testing Five Random Number Generators 206
A.3.1 Serial Correlation Test 206
A.3.2 Uniformity Test 206
A.3.3 K-Dimensional Uniformity Test 208
A.3.4 Runs Up and Down 208
A.3.5 Runs Above and Below the Median 208

A.3.6 Extreme Values 208
A.4 Discussion 214

B Exploratory Examples Using Runs **215**
B.1 Distribution of Total Number of Runs Above and Below Thresholds 215
B.2 Comparison of Different Variogram Functions 217
B.3 Maps of Frequencies of Length of Runs Above Each Threshold . . . 221
B.4 Maps of Differences in Frequencies of Lengths of Runs 221

C Calculation of Uncertainty in the Variogram **229**
C.1 Introduction 229
C.2 Pointwise Variogram Uncertainty 230
C.3 Theoretical Approach 232
C.4 Simulation Alternative 233
C.4.1 Local Simulation Method 233
C.4.2 Global Simulation Method 233
C.5 Validation of Theoretical Approach by Simulation 234
C.6 Example 1: Cluster.dat 234
C.7 Example 2: Red.dat 235
C.8 Transferring Pointwise Uncertainty into the Joint Model 238
C.9 Comments 240

D `HISIM`: Hierarchical Indicator Simulation **243**
D.1 Introduction 243
D.2 The Original Idea 243
D.3 Proposed Approaches 246
D.3.1 Modifying the Mean in Simple Kriging 246
D.3.2 `SIS` Hierarchical 246
D.3.3 Nested Indicator Simulation 247
D.3.4 Correcting Proportions: Markov and Empirical Approaches . 248
D.3.5 Median Hierarchical Indicator Simulation 249
D.4 Conclusions 252

List of Figures

1.1 Example runs in a drillhole. 3

2.1 Example of calculation of a third order covariance. 14
2.2 Forward and downward correction for order relation deviations. . . . 29
2.3 Power model for cumulative distribution function interpolation and extrapolation. 30
2.4 Hyperbolic model for cumulative distribution function extrapolation. 31

3.1 Coding a continuous variable into indicators for different thresholds. 48
3.2 Multiple-point configurations valid as runs and other possible configurations 49
3.3 Increasing conditioning in the calculation of the joint probability of having a run of length L. 54
3.4 Theoretical and experimental results for the calculation of the probability of having a run of length L. 55
3.5 Schematic of hierarchical indicator simulation of runs. 60
3.6 The concept of "accumulated runs". 61
3.7 The histogram of accumulated runs given three runs of length 3, 2, and 2. 61
3.8 Histograms of runs above and below the threshold at the beginning of the simulation. 62
3.9 Function $f(l)$ used in the calculation of the selection function value for each candidate node to be switched. 64
3.10 Impact of switching a node on histograms of runs 66
3.11 The concept of alternating to converge to the solution 69
3.12 Reproduction of runs above and below the median for a random sequence. 70

3.13 Reproduction of runs above and below the median for a regular sequence. 70
3.14 Reproduction of runs above and below the median for a binary array obtained by truncating a multi-Gaussian sequence. 71
3.15 Reproduction of runs above and below the median for a binary array obtained from a realistic exhaustive data set. 71
3.16 Reproduction of runs above and below the median for a binary array obtained from a second realistic exhaustive data set. 72
3.17 Indicator maps for a spatially uncorrelated variable. 73
3.18 Random case: indicator maps for a simulated model using a maximum length of runs of 3. 74
3.19 Random case: indicator maps for a simulated model using a maximum length of runs of 8. 75
3.20 Random case: indicator variograms for a simulated model using a maximum length of runs of 3. 76
3.21 Random case: indicator variograms for a simulated model using a maximum length of runs of 8. 76
3.22 Random case: maps of the training image and the simulated models with maximum length of 3 and 8. 77
3.23 Indicator maps for a multivariate Gaussian correlated variable. . . . 78
3.24 Multi-Gaussian case: indicator maps for a simulated model using a maximum length of runs of 4. 79
3.25 Multi-Gaussian case: indicator maps for a simulated model using a maximum length of runs of 8. 80
3.26 Multi-Gaussian case: indicator variograms for a simulated model using a maximum length of runs of 4. 81
3.27 Multi-Gaussian case: indicator variograms for a simulated model using a maximum length of runs of 8. 81
3.28 Multi-Gaussian case: maps of the training image and the simulated models with maximum length of 4 and 8. 82
3.29 Indicator maps for real data. 83
3.30 Case with real data: indicator maps for a simulated model. 84
3.31 Case with real data: indicator variograms for a simulated model. . . 85

3.32 Case with real data: maps of the training image and the simulated model. . . . 85

3.33 CPU time required to run a model with nine thresholds and considering four directions for the multiple-point runs. . . . 86

4.1 Example of events **A**, **B**, and **C** used in updating techniques 97

4.2 Multiple-point patterns with adjacent grid nodes 99

4.3 Multiple-point patterns extracted from drillhole or well data 99

4.4 Maps of simulated values for small isotropic objects. Proportion above the threshold is 10 % 101

4.5 Maps of simulated values for small isotropic objects. Proportion above the threshold is 50 % 101

4.6 Maps of simulated values for small isotropic objects. Proportion above the threshold is 90 % 102

4.7 Maps of simulated values for large isotropic objects. Proportion above the threshold is 10 % 102

4.8 Maps of simulated values for large isotropic objects. Proportion above the threshold is 50 % 103

4.9 Maps of simulated values for large isotropic objects. Proportion above the threshold is 90 % 103

4.10 Maps of simulated values for small anisotropic objects. Proportion above the threshold is 10 % 104

4.11 Maps of simulated values for small anisotropic objects. Proportion above the threshold is 50 % 104

4.12 Maps of simulated values for small anisotropic objects. Proportion above the threshold is 90 % 105

4.13 Maps of simulated values for large anisotropic objects. Proportion above the threshold is 10 % 105

4.14 Maps of simulated values for large anisotropic objects. Proportion above the threshold is 50 % 106

4.15 Maps of simulated values for large anisotropic objects. Proportion above the threshold is 90 % 106

4.16 Maps of simulated values for a binary image taken from a continuous variable. Proportion above the threshold is 50 % 107

4.17 Maps of continuous simulated values for a continuous variable, using ten thresholds 108
4.18 Histograms of order relation corrections in `SIS` 110
4.19 Indicator variogram before and after correcting for bias due to order relations in `SIS` 111
4.20 Mismatch in MP probability for all 81 MP configurations for four of the methods. 113
4.21 Absolute value of the mismatch in MP probability for all 81 MP configurations for four of the methods. 114
4.22 Graphs of $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ given $P(\mathbf{A})$ under the assumption of full independence and permanence of ratios 116
4.23 Graphs showing the area where the estimated probability $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ is outside the range defined by $P(\mathbf{A}|\mathbf{B})$ and $P(\mathbf{A}|\mathbf{C})$ under the assumption of full independence and permanence of ratios 117

5.1 Histogram of copper grade considering all composites and only composites with rock type code 20 and under elevation 3928 121
5.2 Probability plots for the entire copper grade dataset and for the samples in rock type 20 and under elevation 3928 121
5.3 Projection over the three planes horizontal, vertical along the East-West direction, and vertical along the North-South direction, showing the drillhole data. 122
5.4 Plan views showing the drillhole information. 123
5.5 Plan views showing the locations of drillhole samples with rock type code 20 and samples with other codes 124
5.6 Histogram and lognormal probability plot of copper grade from the blastholes. 125
5.7 Plan views showing the blasthole information. 126
5.8 Histogram and lognormal probability plot of copper grade from the blastholes of benches 3910 and 3922. 127
5.9 Cell size versus declustered mean 127
5.10 Histogram of declustered copper grade from the drillhole data with rock type code 20 and elevations below 3928. 128
5.11 Q-Q plot of drillhole copper values and blasthole sample values. . . . 129

5.12 Cross plots of paired samples for different tolerance distances. 130
5.13 Local mean and variance along the East-West direction of the drillhole and blasthole data. 131
5.14 Local mean and variance along the North-South direction of the drillhole and blasthole data. 132
5.15 Local mean and variance with elevation of the drillhole and blasthole data. 133
5.16 Indicator variogram models 137
5.17 Change in nugget effect and sill contributions for different thresholds. 138
5.18 Change in ranges for different thresholds. 138
5.19 Indicator values of the scattered blasthole data approximated by a regular grid. 139
5.20 Maps of the two benches for the first two realizations by `SIS`. 141
5.21 Histogram and q-q plot of all the simulated values by `SIS` 142
5.22 Histograms of the means and variances of the realizations by `SIS` . . 142
5.23 Q-Q plots of the reference distribution versus the distribution from the first six simulated models by `SIS`. 143
5.24 Cross plot of sample values and the value assigned at the closest node in the models simulated by `SIS` 144
5.25 Definition of the directions for variogram calculation in the regular grid of the model. 145
5.26 Indicator variogram reproduction for direction N30°W (`SIS`). 146
5.27 Indicator variogram reproduction for direction N60°E (`SIS`). 147
5.28 Histogram and q-q plot of all the simulated values under the assumption of independence of the sources of information 149
5.29 Histograms of the means and variances of the realizations obtained by updating under the independence assumption 149
5.30 Histogram and q-q plot of all the simulated values under the assumption of independence of the sources of information 150
5.31 Histograms of the means and variances of the realizations under the assumption of independence of the sources of information 150
5.32 Histogram and q-q plot of all the simulated values under the assumption of permanence of ratios 153

5.33 Histograms of the means and variances of the realizations under the assumption of permanence of ratios 153

5.34 Histogram and q-q plot of all the simulated values under the assumption of permanence of ratios 154

5.35 Histograms of the means and variances of the realizations under the assumption of permanence of ratios 154

5.36 Maps of the two benches for the first two realizations under the assumption of permanence of ratios. 155

5.37 Q-Q plots of the reference distribution versus the distribution from the first six simulated models under the assumption of permanence of ratios. 156

5.38 Indicator variogram reproduction for direction N30°W under the assumption of permanence of ratios. 158

5.39 Indicator variogram reproduction for direction N60°E under the assumption of permanence of ratios. 159

5.40 Histogram and q-q plot of all the simulated values under the multi-Gaussian assumption before correcting for inconsistency between univariate distributions 160

5.41 Histograms of the means and variances of the realizations under the multi-Gaussian assumption before correcting for inconsistency between univariate distributions 161

5.42 Histogram and q-q plot of all the simulated values under the multi-Gaussian assumption after correcting for inconsistency between univariate distributions 162

5.43 Histograms of the means and variances of the realizations under the multi-Gaussian assumption after correcting for inconsistency between univariate distributions 162

5.44 Maps of the two benches for the first two realizations under the multi-Gaussian assumption. 163

5.45 Q-Q plots of the reference distribution versus the distribution from the first six simulated models under the multi-Gaussian assumption. 164

5.46 Indicator variogram reproduction for direction N30°W under the multi-Gaussian assumption. 165

5.47 Indicator variogram reproduction for direction N60°E under the multi-Gaussian assumption. 166
5.48 Normal scores variogram model. The continuous line corresponds to the vertical direction, the dashed line is in the N30°W direction, and the dotted line corresponds to the N60°E direction. 169
5.49 Maps of the two benches for the first two realizations using sequential Gaussian simulation. 170
5.50 Histogram and q-q plot of all the simulated values by `SGS` 171
5.51 Histograms of the means and variances of the realizations by `SGS` . . 171
5.52 Q-Q plots of the reference distribution versus the distribution from the first six simulated models by `SGS`. 172
5.53 Variogram of normal scores reproduction for directions N30°W and N60°E (`SGS`). 173
5.54 Histograms of correlation coefficients between the blasthole data and the closest simulated value 174
5.55 Area considered for calculation of quantity of metal. 175
5.56 Experimental variogram of Cu grades and model used for ordinary kriging. 176

A.1 Schematic showing how `acorn` generates random numbers. 201

B.1 Histograms of total number of runs for different thresholds 216
B.2 Mean and standard deviation of total number of runs for `mcorn` and `acorni` 218
B.3 Mean and standard deviation of total number of runs for sequences with a triangular variogram function generated by moving average . 219
B.4 Mean and standard deviation of total number of runs for sequences with a triangular variogram function generated by simulated annealing220
B.5 Mean and standard deviation of total number of runs for sequences with a range of 5 and different variogram functions 222
B.6 Variogram models used in the examples 223
B.7 Map of frequency of lengths of runs above quantiles for random sequences generated with `acorni` and `mcorn` 223
B.8 Map of frequency of lengths of runs above quantiles for sequences generated by moving average (triangular variogram model) 224

B.9 Map of frequency of lengths of runs above quantiles for sequences generated by simulated annealing (triangular variogram model) . . . 225
B.10 Map of differences of frequencies of lengths of runs above quantiles for sequences generated with `acorni` and `mcorn`. 226
B.11 Map of differences of frequencies of lengths of runs above quantiles for sequences generated by moving average 227
B.12 Map of differences of frequencies of lengths of runs above quantiles for sequences generated by simulated annealing 228

C.1 Calculation of fourth order covariances $C_{ij}(\mathbf{h})$ 231
C.2 Location map of samples taken from Cluster database. 234
C.3 Experimental and fitted variogram, and central confidence intervals at 95 %, 75 %, 50 %, and 25 % for each lag (Cluster database) . . . 235
C.4 Location map of samples and gold content taken form the Red database.236
C.5 Experimental and fitted variogram, and central confidence intervals at 95 %, 75 %, 50 %, and 25 % for each lag (Red database) 236
C.6 Experimental variogram values calculated using all simulated data and only the simulated values at sampling locations (Red database) 237
C.7 An example of the uncertainty distribution of the pointwise variogram values 239
C.8 An example of an incorrect interpretation of joint uncertainty given the pointwise uncertainty 240
C.9 An example of a correct interpretation of joint uncertainty 240

D.1 Map showing the result for the original implementation of `HISIM` . . 244
D.2 Variogram reproduction for the original implementation of `HISIM`. . . 245
D.3 `HISIM` varying the simple kriging mean for a single threshold case . . 246
D.4 `SIS` applied hierarchically 247
D.5 Hierarchical application of `SIS` using a Markov assumption 249
D.6 Empirical adjustment of the proportion to apply `SIS` hierarchically . 250
D.7 Illustration of median hierarchical indicator simulation 250
D.8 Illustration of the case when drawing nodes by Monte Carlo simulation is virtually random and when the drawing is effective and accounts for the differences in probabilities 252

D.9 Application of median hierarchical indicator simulation for an intrinsically correlated variable or mosaic model. 253
D.10 Application of median hierarchical indicator simulation for a multi-Gaussian variable. 254
D.11 Application of median hierarchical indicator simulation for a non-Gaussian variable. 255

List of Tables

4.1 Mean squared error and mean absolute error in the MP probability for the different algorithms. 112

4.2 Fraction of the nodes updated where $P(\mathbf{A}|\mathbf{B}, \mathbf{C})$ was outside the range of $P(\mathbf{A}|\mathbf{B})$ and $P(\mathbf{A}|\mathbf{C})$. 115

5.1 Threshold definition for indicator variogram calculation and simulation 133

5.2 Parameters for calculation of experimental variograms 134

5.3 Indicator variogram model parameters. 135

5.4 Grid definition for multiple-point inference and simulation. 136

5.5 Simulation parameters. 140

5.6 Summary of order relation deviations for a particular realization in `sis`. 145

5.7 Summary of order relation deviations for a particular realization, before correcting for inconsistency of univariate distributions, under the assumption of independence of the sources of information. 151

5.8 Summary of order relation deviations for a particular realization, after correcting for inconsistency of univariate distributions, under the assumption of independence of the sources of information. 152

5.9 Summary of order relation deviations for a particular realization under the assumption of permanence of ratios, before correcting for inconsistency of univariate distributions. 157

5.10 Summary of order relation deviations for a particular realization under the assumption of permanence of ratios, after correcting for inconsistency of univariate distributions. 160

5.11 Summary of order relation deviations for a particular realization under the multi-Gaussian assumption, before correcting for inconsistency of univariate distributions. 167

5.12 Summary of order relation deviations for a particular realization under the multi-Gaussian assumption, after correcting for inconsistency of univariate distributions. 167
5.13 Normal scores variogram model parameters. 168
5.14 Simulation parameters. 169
5.15 Expected quantity of metal based on the different methods, compared to the "truth" computed by ordinary kriging of the blastholes. . . . 176

A.1 Results of serial correlation test 207
A.2 Results of uniformity test 209
A.3 Results of k-dimensional uniformity test 210
A.4 Results of runs up and down test 211
A.5 Results of runs above and below the median test 212
A.6 Results of maximum values test 213

B.1 Theoretical and observed results - `mcorn`. 215

C.1 Pointwise variogram uncertainty calculated using the three methods presented. 235
C.2 Theoretical approach to calculate the variogram confidence intervals. 238

Chapter 1

Introduction

1.1 Problem Setting

Geostatistical simulation provides tools to quantify uncertainty to help in decision making.

For example, mine planning and scheduling could be done with a set of realizations, that is, numerical models of properties such as specific gravity, rock type, and grades that share some characteristics of the real mineralization. Planning could ensure that the mill would have a guaranteed tonnage and grade for each production period.

In petroleum, several response variables are considered: hydrocarbon recovery, breakthrough time, and flow rate. Several recovery schemes can be proposed and the response variables evaluated through multiple realizations by flow simulation. A histogram of possible responses is then generated to help decide the best recovery scheme.

When considering environmental applications, it is important to have an assessment of the uncertainty in, for example, the concentration of a pollutant at different locations. Good reproduction of the features of the real phenomenon will allow accurate estimation of the probability of exceeding a regulatory threshold.

In all cases, the transfer functions –mine plan, flow simulation, or compliance with a threshold– are highly sensitive to the existence of long range connectivity, that is, paths or patches of high or low values. Geostatistical realizations should correctly reproduce these important aspects of the true distribution.

Classical geostatistical simulation techniques such as Gaussian and indicator approaches account for only two-point statistics through a covariance or variogram function. This limitation is mainly due to the difficult inference and modelling of higher order statistics. The integration of information of different types is also a complicated problem: the redundancy between different sources must be accounted for and, in practice, it is not easy to quantify. Simulation techniques would be improved with multiple-point statistics, since the resulting realizations would share more quantitative information with the underlying true distribution. More realistic numerical models will surely lead to better decisions.

The implementation of simulation algorithms that account for multiple-point statistics must overcome two main problems: (1) inference of the multiple-point statistics, and (2) development of an algorithm that integrates multiple-point statistics into simulation.

Inferring multiple-point statistics is difficult because the information available is scarce. The spatial configuration must be repeated several times to estimate different moments of the multivariate distribution, that is, the probability of having different combinations of the values or classes in the same spatial arrangement. In practice, it is impossible to have all spatial configurations of multiple-points and combinations of values to infer probabilities. This is one of the reasons to use training images to supplement the data. Nevertheless, a few typical patterns are available from the data. If drillholes are drilled in the same direction, several similar combinations of data are available. The grade may be coded as an indicator at some critical threshold. The number of multiple point events is largely reduced. In the case of three points, there would be $2^3 = 8$ combinations, that is, $\{1,1,1\}$, $\{1,1,0\}$, $\{1,0,1\}$, $\{1,0,0\}$, $\{0,1,1\}$, $\{0,1,0\}$, $\{0,0,1\}$, and $\{0,0,0\}$. If blastholes are available, two dimensional multiple-point patterns can easily be extracted. In practice, blastholes are spaced in a pseudo-regular grid, hence the inference of the probability of having a given multiple-point configuration becomes possible with some minor approximations regarding the exact location of the points of the pattern.

Several researchers have tried to incorporate multiple-point statistics in simulation. Most of them have not been adopted in practice, because they are extremely CPU time consuming or because they require too many parameters to be set.

Training images have been used to extract multiple-point statistics from outcrops or from conceptual geological models. Although this is a valid approach to obtain multiple-point statistics, a data-driven approach to modelling is preferred and these statistics are extracted from data.

Some interesting results in number theory motivate us to study application of the indicator approach. This research aims to explore methods to incorporate multiple-point information extracted from data, into geostatistical simulation. Training images are often used to test the methods.

Multiple-point statistics extracted from data are not used in existing multiple-point simulation methods. The applicability of the methods proposed in this research depends on drillhole and blasthole data, where the number of replications of multiple-point configurations is enough to calculate the expected probabilities of these configurations happening.

Two types of patterns are being considered, although the methodologies can be extended to any pattern, if abundant data are available. The first type of multiple-point configuration is one-dimensional. It is what is called a *run*. To explain the concept, **Figure 1.1** presents a drillhole with 22 composites (samples of equal length). The actual grade is shown as a solid line, while the sample values are shown as black dots. A run of length l above a threshold can be seen as the event of having l consecutive samples with grade higher than the threshold. In the example of **Figure 1.1** runs are represented by thicker solid lines. For z_1, there is one run of length 16; for z_2, there is one run of length 13; for z_3, there are two runs, one of length 4 and the other of length 5; for z_4 there are three runs of lengths 2, 2, and 1 respectively; finally, for z_5 there are no runs. The probability of having a run of length l is equivalent to the expected value of the product of l indicators corresponding to adjacent samples, or equivalently, to their non-centered l-order indicator covariance. This high order moment is a multiple-point statistic that characterizes the

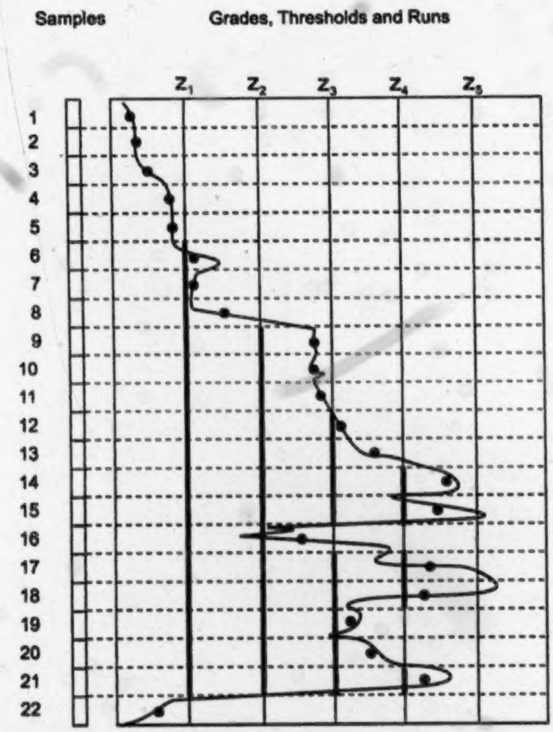


Figure 1.1: Example runs in a drillhole with 22 samples. The black dots are the sample values. The runs are presented as thick solid lines under each threshold $z_i, i = 1, ..., 5$.

true multivariate distribution or spatial law. Notice that the spatial law cannot be *fully* characterized only by runs, but the information provided by these statistics is more complete than the one obtained by simply using the variogram.

A second type of data corresponds to two-dimensional configurations obtained from samples taken at blastholes. Since blastholes are generally regularly spaced, enough replications of the same configuration can be found. If the grades are coded as indicators for a given cutoff, then inference of multiple-point statistics is possible.

Two different methodologies have been explored with mixed success:

- A methodology to honor the frequency of multiple-point runs was developed, that is, the probability of having a number of adjacent nodes with the same indicator value. The difficulty in finding a rule to define the nodes that are above or below the threshold is illustrated.

- A different approach was also developed to integrate the multiple-point information into the traditional sequential indicator simulation framework, while still honoring histogram and indicator variograms. Three different assumptions about the redundancy between the variogram or covariance function and the multiple-point statistics extracted from the data are discussed:

    1. The assumption of independence of the information provided by several

sources to estimate the indicator values at an unsampled location.

2. The assumption of permanence of ratios, that is, the incremental information provided by one source is constant before and after knowing the information provided by the other sources.
3. A multi-Gaussian approximation of the relationship between the different sources of information.

Some of the drawbacks of the techniques are exposed and the methods are implemented and tested with a real data set from a porphyry copper mine.

1.2 Proposed Approach

1.2.1 Accounting for Multiple-Point Statistics as Runs

A method to simulate a continuous variable using the indicator framework, and using multiple-point statistics in the form of runs, in a hierarchical fashion is first explored.

Given histograms of frequencies of runs in several directions, for different thresholds, the algorithm starts with the highest threshold and erodes the initially high valued field, to account for the histograms of runs above and below that threshold in multiple directions. All nodes are initially coded as 0 (above the threshold). Once enough nodes have been switched to 1, that is, they are set to be below the threshold, the simulation at the current threshold stops. The nodes that still have a value of 0 correspond to high values. These are simulated using some extrapolation function, as it is usually done in indicator simulation. The remaining nodes, the ones that have their indicators set to 1 are reset to 0 for the next (lower) threshold, and the algorithm erodes this new constrained domain, until enough nodes have been switched to 1. Now, the nodes with an indicator set at 0 are valued between the current threshold and the previous (higher) threshold. This hierarchical procedure is repeated until all thresholds have been simulated. It can be seen as a simulation into consecutively constrained domains that are nested within another.

The decision to switch a node is based on the favorableness of that change to converge to the histogram of runs above and below that particular threshold. A decision rule is applied that permits this convergence.

As with conventional indicator simulation, interpolation and extrapolation beyond the discretized local distribution of uncertainty is required to draw a value in a continuous domain.

The algorithm does not require direct input of the variogram or indicator variograms. The histogram is reproduced by construction. Enough nodes are switched at each threshold to reproduce the global distribution.

This method generates artifacts that invalidate the results for practical application, although it opens an area of research that has not been explored. The definition of the decision rule to switch nodes to be above or below the threshold seems to be key to ensure convergence and to avoid artifacts.

Several examples are presented to illustrate the implementation and problems of this methodology.

1.2.2 Integrating the Indicator Kriging Probability and the Multiple-Point Statistics under Different Assumptions

A second approach is to integrate the indicator kriging probability and high-order statistics under some assumption of their relationship. This integration requires the knowledge of the redundancy between both statistics, which implies that the knowledge of the multivariate spatial distribution is required. This is only possible when using a training image to extract the two-point and multiple-point statistics or under the parametric multi-Gaussian model. Some approximations are proposed for the general case, where the multivariate distribution is unknown. Three different assumptions regarding the relationship between the different sources of information are discussed:

1. Assuming independence between the probability estimated by indicator kriging and the multiple-point probability allows a simplification of the expression for their joint probability, obtained through Bayes' law. This assumption appears as unrealistic in a spatial context and its implementation carries serious difficulties due to the possibility of generating values for the probability in excess of one.

2. Assuming that the incremental information provided by one source is constant regardless of the additional available information from other sources also permits an expression to be obtained for the joint probability between several sources. This assumption of permanence of ratios is implemented without major difficulties and shows an improvement in the performance of the numerical models when applied for medium or long term planning in a mine.

3. Assuming the relationship between the several sources of information is multivariate-Gaussian, the redundancy between them can be assessed. A new estimate of the indicator value for a given threshold can be built by linearly combining the probabilities coming from indicator kriging and multiple-point statistics. The assumption allows the determination of the weights assigned to each probability. The implementation of this methodology is straightforward, although the results do not show any improvement with respect to the standard indicator simulation method.

A practical case study is presented to illustrate the methodologies. Advantages and drawbacks of each method are discussed.

1.3 Dissertation Outline

Chapter 2 discusses the theoretical basis used in this dissertation, by providing an overview of geostatistical methods. The concepts of spatial law and multivariate spatial distribution are explained. Problems encountered when inferring spatial statistics are discussed. Conventional estimation and simulation methods that account for two-point statistics are reviewed as well as the attempts made to incorporate multiple-point statistics in simulation.

Chapter 3 presents the methodology proposed to infer multiple-point statistics as runs from drillhole or well data, and the implementation of the hierarchical simulation of runs based on the indicator approach. Problems encountered are discussed.

Chapter 4 is devoted to methods that integrate the multiple-point statistics into the sequential indicator simulation framework. The multiple-point information is considered as a secondary source of information and the relationship between the two-point and multi-point statistics is approximated under different assumptions.

Chapter 5 shows a case study using actual data from a producing mine.

Finally, discussion on the issues encountered in the application of the proposed methods, as well as future work and conclusions are presented in **Chapter 6**.

The thesis includes several appendices with results from related studies. **Appendix A** shows a review on random number generators and tests for high-order correlation. **Appendix B** presents results from different exploratory exercises computed to better understand the behavior of multiple-point statistics as runs. Changes in these statistics due to the choice of the algorithm that constraints only up to the second-order (the variogram) are illustrated with several examples. **Appendix C** shows an application where multiple-point statistics are required to calculate the uncertainty on the variogram. The parametric multi-Gaussian distribution is used to overcome the problem of inferring these statistics. Finally, **Appendix D** gives the background for the hierarchical method implemented in Chapter 3 without considering the multiple-point statistics, but only the indicator variograms.

Chapter 2

Overview of Geostatistics

This chapter presents an overview of the theoretical background necessary to proceed to modelling accounting for multiple-point statistics. Many books that contain some of the topics presented in this chapter are available and can be used as a source for additional information on the concepts reviewed here (see [20, 41, 43, 82, 94]).

Geostatistics deals with the prediction of variables distributed in space. It uses the spatial correlation to quantify the relationship of the values of the variable taken at different locations. This spatial correlation is often calculated using two points at a time. As a branch of applied statistics, geostatistics works under a probabilistic framework that allows inference. Numerical models are constructed to estimate the value of the variable and to simulate its spatial features for uncertainty quantification. In **Section 2.1**, basic concepts of the Theory of Regionalized Variables are introduced.

Definitions for univariate, bivariate and multivariate moments and a discussion on statistical inference are presented in **Section 2.2**.

Estimation is done by kriging, which corresponds to linear regression in a spatial context, that is, taking into account the dependence among the data. The variable at an unsampled location is predicted with the information provided by a set of samples within a neighborhood. The estimate is built as a linear combination of the data values, although some non-linear estimators also exist. The weights assigned to each sample are chosen to minimize the mean squared error calculated between the estimated value and the true one. A brief presentation of estimation techniques is provided in **Section 2.3**.

Simulation is done to assess performance considering the joint variability of petrophysical properties such as concentration of elements, porosity or permeability. Uncertainty in response variables can be quantified. These response variables can be as simple as a block average or as complex as a mine schedule for production. Kriging estimates are smooth and do not reproduce the variability of the true variable. Multiple realizations can be built through simulation that honor the sample data, the representative histogram, and the spatial correlation known as the variogram. Conventional simulation techniques are described in **Section 2.4**.

Additional features can be injected by considering more than just the variogram. These consider the relationship between more than two points at the same time and are known as multiple-point statistics. Some methods have been proposed to account for these statistics, however they have not been widely applied. A review of the approaches that incorporate multiple-point statistics is presented in **Section 2.5**.

The chapter ends in **Section 2.6** with a brief discussion on the reasons for limited applicability of algorithms that account for multiple-point statistics.

2.1 The Theory of Regionalized Variables

In 1965 Georges Matheron introduced the Theory of Regionalized Variables [117], formalizing notions that had been around for a few decades in various fields [58, 105, 115, 116, 150]. Journel and Huijbregts [94] summarize this theory. The following presentation of the Theory of Regionalized Variables is based mainly on Journel and Huijbregts.

Natural phenomena can be characterized by measuring one or more variables distributed in space. Those variables are called *regionalized variables* and the underlying phenomenon, a *regionalization*. A measure of the value of interest, z, can be performed at any location $\mathbf{u}$ in space. The set of all the measures in the domain of interest $\{z(\mathbf{u}), \mathbf{u} \in D\}$ represents the "reality", which is unknown in practice. Typical examples of regionalized variables are the copper grade of a composite of length L, the thickness of a gold vein, and the depth of a seam on a coal mine.

The definition of a regionalized variable does not carry any probabilistic interpretation *per se*; however, when observing measurements of regionalized variables, two characteristic features are seen: they present an apparent local random behavior and a general structured aspect. For example, when looking at copper grades in a porphyry deposit, the following structured behavior of the grades can be found: the closer two samples are, the more similar their grades. On the other hand, a random behavior is also observed: two samples very close to each other could have grades that differ in an unpredictable manner. The concept of a *random variable* is then introduced. A random variable is a variable that can take a value according to a probability distribution. For example, in a copper deposit, the grade obtained in a sample at a given location $z(\mathbf{u})$ is assumed to be drawn from a probability distribution $f_Z(z)$ and is seen as a particular realization of the random variable $Z(\mathbf{u})$. Although, only one true grade exists at that location, this probabilistic approach is taken to handle the ignorance regarding the grade at unsampled locations.

The set of all random variables in the domain is called *random function*.

$$\text{Random Function} \sim \{Z(\mathbf{u}), \forall \mathbf{u} \in D\}$$

Each of the random variables $Z(\mathbf{u})$ has a probability distribution and they are related with each other. The set of all joint distributions or multivariate spatial distributions $f_{Z(\mathbf{u}_1),Z(\mathbf{u}_2),\ldots,Z(\mathbf{u}_k)}$ for any finite integer k and all locations $\mathbf{u}_i \in D, i = 1,\ldots,k$ corresponds to the *spatial law* of the random function. It can be seen as the simultaneous behavior of several points at the same time or their joint variability. The spatial law characterizes the dependence of all these multiple points.

This idea can also be extended to multiple variables. Consider the case of three different concentrations of interest in a deposit. These three variables can be measured at a location $\mathbf{u}$, generating three values $z_1(\mathbf{u})$, $z_2(\mathbf{u})$, and $z_3(\mathbf{u})$. When considering all three random variables at all locations, this becomes the random function.

The three values have their own spatial structure and they may be cross-correlated, for example, the value of variable Z_1 taken at location $\mathbf{u}_1$ may be informative to estimate the value of Z_2 at the same or different location $\mathbf{u}_2$. A typical case is when different elements are deposited by the same mineralizing event. Their concentrations will likely be correlated, that is, when the concentration of one element increases in one location, it is probable that the concentration of the second element will also be higher there. For example, it is common to find that the concentration of molybdenum is positively correlated with copper in porphyry type deposits.

The spatial law can be characterized by several of its moments. For example, the mean, variance, and spatial correlation. Since in general the spatial law is unknown, these moments must inferred from the data. The more moments are specified, the more detailed information is available of the multivariate spatial distribution of the variable. This will allow more accurate inference of the probability distributions at unsampled locations, which will lead to better informed decisions.

The goal of this probabilistic interpretation is to allow inference at unsampled locations. The probability distribution is a model of the lack of knowledge regarding the value of the regionalized variable at that location. A measurement cannot be repeated at the same location $\mathbf{u}_1$ to "sample" the probability distribution of the random variable $Z(\mathbf{u}_1)$. However, under some assumption of *stationary*, taking samples at different locations $\mathbf{u}_2, \mathbf{u}_3, \ldots, \mathbf{u}_k$ provides a prior model of the probability distribution for $Z(\mathbf{u}_1)$. Therefore, inference of moments of the probability distribution calls for this assumption of stationarity (refer to **Section 2.2** below).

The convention of using upper case to denote random variables and random functions is followed, for example, $Z(\mathbf{u})$. The regionalized variable and its actual values are denoted in lower case, $z(\mathbf{u})$.

2.2 Statistical Inference and Stationarity

Inference is possible because of the random function formalism. The actual values of the regionalized variable are seen as a realization of a set of random variables (the random function). The decision of stationarity defines the data that are pooled together for statistical inference [88].

Given a set of samples $\{z(\mathbf{u}_\alpha), \alpha = 1, ..., N\}$, inference of some of the moments of the population is required. These moments are inferred using the experimental frequencies calculated from the data or from some secondary source of information, such as a training image. Those statistics are then used as input in estimation and simulation algorithms.

A very basic mathematical operator is the *expected value* of a random variable. The expected value of a function of a random variable $g(Z)$ is:

$$E\{g(Z)\} = \begin{cases} \int_{-\infty}^{\infty} g(z) f_Z(z) dz & \text{if } Z \text{ is continuous} \\ \sum_{z \in \chi} g(Z) P(Z = z) & \text{if } Z \text{ is discrete} \end{cases}$$

provided that the integral or sum exists. χ represents the domain of the categorical values of Z in the discrete case. Notice that if Z is a random variable, then any function of Z, in this case $g(Z)$, is also a random variable.

The expected value has several properties that are useful to determine relationships between different moments of the random variable as discussed next [19].

2.2.1 Moments of a Random Variable

The first moment of a random variable $Z(\mathbf{u})$, called the *mathematical expectation*, is defined as:

$$E\{Z(\mathbf{u})\} = m(\mathbf{u})$$

Notice that in general, the mathematical expectation depends on the location $\mathbf{u}$.

Second-order moments can be considered between any two points in space. The *variance* is defined at a given location $\mathbf{u}$ as the second moment around the mean. It is also, in general, a function of the location $\mathbf{u}$:

$$Var\{Z(\mathbf{u})\} = E\{(Z(\mathbf{u}) - m(\mathbf{u}))^2\}$$

The *standard deviation* corresponds to the square root of the variance. It has the advantage of being in the same units as the variable.

When considering two different points in space, $\mathbf{u}_1$ and $\mathbf{u}_2$, the *centered covariance* is calculated as:

$$Cov\{\mathbf{u}_1, \mathbf{u}_2\} = E\{(Z(\mathbf{u}_1) - m(\mathbf{u}_1)) \cdot (Z(\mathbf{u}_2) - m(\mathbf{u}_2))\} \tag{2.1}$$

Notice that when $\mathbf{u}_1 = \mathbf{u}_2$, the covariance becomes the variance.

Another second-order moment is the *semi-variogram*, defined as half the variance of the difference between the variable at two different locations:

$$\gamma(\mathbf{u}_1, \mathbf{u}_2) = \frac{1}{2} Var\{Z(\mathbf{u}_1) - Z(\mathbf{u}_2)\}$$

The prefix *semi-* was used to emphasize that it corresponds to half the variance of the difference $Z(\mathbf{u}_1) - Z(\mathbf{u}_2)$, however, in current literature the semi-variogram is simply called the *variogram*. From now on, the prefix *semi-* will be dropped from the semi-variogram and it will be called *variogram*.

The *correlogram* is defined as the standardized covariance, that is, the covariance divided by the corresponding standard deviations:

$$\rho\{\mathbf{u}_1, \mathbf{u}_2\} = \frac{Cov\{\mathbf{u}_1, \mathbf{u}_2\}}{\sqrt{Var\{Z(\mathbf{u}_1)\} \cdot Var\{Z(\mathbf{u}_2)\}}}$$

These moments can also be defined in a multivariate context.

Cross-covariance: $Cov_{Z_1,Z_2}\{\mathbf{u}_1, \mathbf{u}_2\} = E\{(Z_1(\mathbf{u}_1) - m_{Z_1}(\mathbf{u}_1)) \cdot (Z_2(\mathbf{u}_2) - m_{Z_2}(\mathbf{u}_2))\}$

Cross-variogram: $\gamma_{Z_1,Z_2}(\mathbf{u}_1, \mathbf{u}_2) = \frac{1}{2} E\{(Z_1(\mathbf{u}_1) - Z_1(\mathbf{u}_2)) \cdot (Z_2(\mathbf{u}_1) - Z_2(\mathbf{u}_2))\}$

Cross-correlation: $\rho_{Z_1,Z_2}\{\mathbf{u}_1, \mathbf{u}_2\} = \frac{Cov_{Z_1,Z_2}\{\mathbf{u}_1,\mathbf{u}_2\}}{\sqrt{Var\{Z_1(\mathbf{u}_1)\} \cdot Var\{Z_2(\mathbf{u}_2)\}}}$

The notion of covariance can be extended to multiple-points. Denote Cov_n the *n-point centered covariance*:

$$Cov_n\{\mathbf{u}_1, \dots, \mathbf{u}_n\} = E\{\prod_{i=1}^{n} (Z(\mathbf{u}_i) - m(\mathbf{u}_i))\}$$

Many other moments can be calculated after transforming the data. If a variable $Y = f(Z)$ is considered, the transformed variable Y will have a mathematical expectation, variance, covariance, and high-order moments (see **Section 2.4**).

2.2.2 The Decision of Stationarity

Stationarity is a decision that the data come from the same population and can be used to infer different statistics. Several types of stationarity can be defined [117]. In general it can be said that a random function is *stationary of order n* if all the n^{th} order moments exist and are independent of the location of the points used to calculate the n-order moment, that is, $\forall \mathbf{u}$:

$$\begin{array}{lrcl}
\text{Order 1:} & E\{Z(\mathbf{u})\} & = & m \\
\text{Order 2:} & Cov\{\mathbf{u}, \mathbf{u}+\mathbf{h}\} & = & Cov\{\mathbf{h}\} = E\{Z(\mathbf{u}) \cdot Z(\mathbf{u}+\mathbf{h})\} - m^2 \\
\quad \ldots & \ldots & & \\
\text{Order n:} & Cov_n\{\mathbf{u}+\mathbf{h}_0, \mathbf{u}+\mathbf{h}_1, \ldots, \mathbf{u}+\mathbf{h}_{n-1}\} & = & Cov_n\{\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_{n-1}\} \\
 & & = & E\{\prod_{i=1}^{n} (Z(\mathbf{u}+\mathbf{h}_{i-1}) - m)\}
\end{array}$$

Notice that none of the statistics above depends on the location of the points, but only on their spatial configuration.

In most geostatistical applications, stationarity up to the second order is of interest. Note that stationarity in the covariance (order 2) implies the existence of the variance and the stationarity of the variogram. Considering the covariance at a lag $\mathbf{h} = 0$, then the definition of the covariance identifies the variance:

$$Cov\{\mathbf{u}, \mathbf{u}+\mathbf{h}\} = Cov\{\mathbf{0}\} = Var\{Z\{\mathbf{u}\}\}$$

Under *second order stationarity*, the variogram can be related to the covariance and the variance:

$$\gamma(\mathbf{h}) = Cov\{\mathbf{0}\} - Cov\{\mathbf{h}\}$$

Some phenomena show an apparent infinite capacity of dispersion and therefore the variance and covariance cannot be calculated [142]. The assumption of second order stationarity may not be correct for the data. *Intrinsic stationarity* is a less constraining condition than second order stationarity. It assumes that the mean exists and that the variogram depends on the spatial configuration, or equivalently that the increments $Z(\mathbf{u}) - Z(\mathbf{u}+\mathbf{h})$ are second order stationary, hence the variogram exists. However, the covariance is not defined.

2.2.3 Inferring Representative Histograms

Sample data are used to construct a global stationary histogram. Statistical measures of the distribution are estimated from the histogram; however, when sampling is spatially biased the histogram must be corrected.

Declustering corrects for preferential sampling [9, 33, 43, 71, 82]. In mining, the high valued zones or the area that will be produced first is of economic importance and will be sampled more closely. Declustering can be performed in several ways:

Polygonal declustering. This method works by volume of influence of each sample. The denser the sampling in a given zone, the less influence each sample will have. One of the problems of this method is how to handle the edges of the domain. If the domain is well delineated, then it gives, in general, reliable results. If, on the other hand, the edges of the domain are not clear, the outer samples will have too much or too little influence. This is especially important when the outer samples are poorer than the ones in the center of the domain and global estimation is required. The mean value of the variable will drop as the domain is made bigger. Several approaches can be used to limit the influence of the outer samples: a radius of influence can be assigned to the samples so that they only inform up to a maximum volume.

Cell declustering. This method handles the problem of the domain boundary. The domain is divided into cells or blocks that receive equal weight; every sample is assigned the same weight within the cell. The more samples in a cell, the less influence they will have in the global statistics. The question is how to pick a cell size. Common practice is to run the algorithm with several cell sizes and select the one that minimizes (or maximizes, if the samples were biased toward the low values) the mean. Although there is no reason to pick the minimum, it has given reasonable results [33].

Kriging. Ordinary kriging weights can be used as a measure of influence (see **Section 2.3** for further discussion on kriging). The samples that have the higher influence in estimating the points in the domain will have higher kriging weights. The sum of the weights assigned to a given sample will be standardized and used as its weight. The advantage of this method for declustering is that it accounts for the configuration of the data and the spatial continuity [82]. However, one feature of kriging is that it assigns larger weights to samples at the end of strings such as drillholes. This will have an impact on the declustering weights [36, 38].

Declustering only changes the weight of the sample values in terms of its probability in the global distribution, but does not change the value itself. Therefore, these techniques will not be able to correct for sampling that did not cover the entire range of the variable. *Debiasing* methods are then required. Two methodologies are available:

Detrending the model. If enough evidence that a trend in the variable exists [97], then a trend model should be constructed and the geostatistical study should be carried on working with residuals. The trend model can be constructed by combining the horizontal and vertical trend, as suggested by Deutsch [41]. An alternative is to work in a framework that implicitly accounts and models the trend, such as intrinsic random functions of order k or universal kriging [20]. Unfortunately, for simulation purposes, departures from the stationarity assumption will greatly affect the final result [41].

Bivariate calibration. If some secondary measurements exist, for example from a geophysical survey, and if the relationship between the variable of interest and this secondary variable is known analytically or experimentally, then this bivariate relationship can be used to correct the histogram of the variable of interest, given exhaustive (or denser) samples of the secondary variable over the domain [138].

2.2.4 Variogram Inference

The variance can be calculated from the corrected histogram. The most important bivariate statistic used in geostatistics is the variogram. Inference of the variogram has been extensively discussed in the literature [3, 4, 20, 21, 43, 47, 61, 71, 75, 94, 129, 130, 134, 156].

The experimental variogram is estimated as half the average of squared differences between data separated exactly by a distance vector **h**. In practice, angle and lag tolerances are defined so that a reasonable number of pairs approximately **h** apart, $n(\mathbf{h})$ can be found:

$$\hat{\gamma}(\mathbf{h}) = \frac{1}{2 \cdot n(\mathbf{h})} \cdot \sum_{i=1}^{n(\mathbf{h})} [z(\mathbf{u}_i) - z(\mathbf{u}_i + \mathbf{h})]^2$$

The number of lags, lag separation distance and tolerances (vertical and horizontal angles and band widths) may help to get a reliable estimate of the variogram, although this is not always possible [25]. Bad choices will generate noisier plots that are not representative of the underlying population.

Variograms must be modelled to be incorporated to estimation or simulation algorithms. Models are considered licit if they are positive-definite, that is if they are a valid measure of distance [5]. The positive-definiteness constraint ensures that the estimation variance will be positive or zero. Otherwise the mathematical model would not be valid since the variance must be non-negative, by definition.

When more than one variable exist cross-variograms measure their relationship in space, that is, how similar the variables at two locations are. The cross-variogram can be calculated as:

$$\hat{\gamma}(\mathbf{h})_{ZY} = \frac{1}{2 \cdot n(\mathbf{h})} \cdot \sum_{i=1}^{n(\mathbf{h})} [z(\mathbf{u}_i) - z(\mathbf{u}_i + \mathbf{h})] \cdot [y(\mathbf{u}_i) - y(\mathbf{u}_i + \mathbf{h})]$$

Modelling variograms and cross-variograms is even more demanding. A valid model of coregionalization is required. This means that direct and cross variogram models must be consistent with each other and provide a measure of spatial correlation that makes physical sense. The positive definiteness condition ensures that when solving a cokriging system the estimation variance is positive (see related note in **Section 2.2.6**).

One assumption regarding cross-variograms is that the correlation is symmetric with respect to the direction of the vector **h**. In some applications, this may not be a correct assumption, and a model that can handle the offset in the correlation may be required. Cross-covariances are more flexible, since they do not require the primary and secondary variables to be measured at the same locations.

Considering the mean of the primary and secondary variables to be known and equal to m_Z, and m_Y respectively, an experimental cross-covariance can be calculated as:

$$\hat{C}_{ZY} = \frac{1}{n(\mathbf{h})} \cdot \sum_{i=1}^{n(\mathbf{h})} [z(\mathbf{u}_i) - m_Z] \cdot [y(\mathbf{u}_i + \mathbf{h}) - m_Y]$$

The order in which the variables are considered in the calculation matters, and a second cross-covariance can be calculated by switching the variables for the head and tail of the lag vector **h** [20].

2.2.5 Inferring Multiple-Point Statistics

Working with multiple-points is more demanding than with only two points at a time. Multiple configurations must be found in space to provide an estimate of the frequency of occurrence of each arrangement of values.

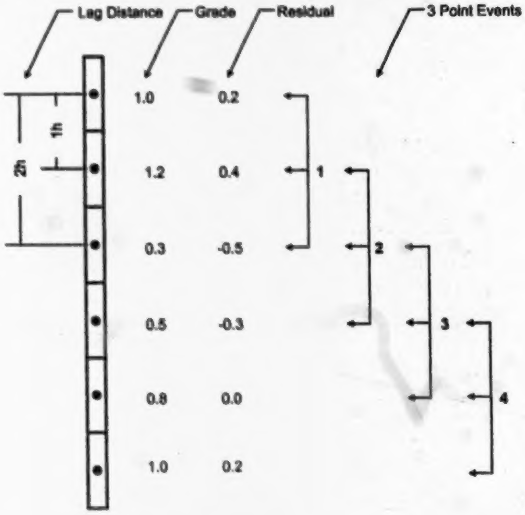


Figure 2.1: Example of calculation of a third order covariance.

Consider for example the calculation of the covariance of third order (**Figure 2.1**) with only one particular lag at a given direction to illustrate how this is done. If there are some composites of length **h** calculated from drillhole data, the composite size can be used as lag separation distance for the calculation. Cov_3 is calculated as the average product of the residual values from the mean separated by the vectors **h** and **2h**. In this example the mean is 0.8:

$$\begin{aligned} Cov_3(\mathbf{h}, 2\mathbf{h}) &= \tfrac{1}{4}((1.0-0.8)\cdot(1.2-0.8)\cdot(0.3-0.8)+(1.2-0.8)\cdot(0.3-0.8)\cdot(0.5-0.8) \\ &\quad +(0.3-0.8)\cdot(0.5-0.8)\cdot(0.8-0.8)+(0.5-0.8)\cdot(0.8-0.8)\cdot(1.0-0.8)) \\ &= \tfrac{1}{4}((0.2)\cdot(0.4)\cdot(-0.5)+(0.4)\cdot(-0.5)\cdot(-0.3) \\ &\quad +(-0.5)\cdot(-0.3)\cdot(0.0)+(-0.3)\cdot(0.0)\cdot(0.2)) \\ &= \tfrac{1}{4}((-0.04)+(0.06)+(0.0)+(0.0)) \\ &= 0.005 \end{aligned}$$

As shown in the previous example, the probabilities of multiple-point events are estimated with their relative frequencies found in the data set or training image. To estimate the third order covariance, "three point events" were used, that is, all the possible combinations of three points that met the requirements defined by the lag separation distances being **h** and **2h**. This can be generalized to any two distance vectors $\mathbf{h}_1$ and $\mathbf{h}_2$. The data can be coded in different ways (see discussion later). Indicator coding is a typical approach, since it reduces the dimensionality of the problem by defining a binary variable, depending on the grade being greater than a cutoff. A multiple-point event can now be defined based on the value and geometric configuration of the points. This notion will be formalized later.

The higher the order of the statistic, that is, the number of points considered at the same time, the larger the number of required samples. It is practically impossible to know the probability of all spatial configurations of n-points.

Inference will only be possible if multiple replications of an event are available to calculate its frequency. In practice most of the samples are taken at drillholes as almost linear strings. The frequencies of low-order statistics, such as the indicator values for strings of 3, 4, or 5 composites in the vertical direction may be possible to infer.

Linear data could not be used to infer curvilinear features. In this case, training images or data in two-dimensional arrangements, such as blasthole data, are required. The use of training images is appealing because the extraction of frequencies of multiple-points events is consistent. The problem of having a positive definite model is resolved when done on a single image. Another problem arises when a given event is not found in the training image. This can be solved by reducing the dimension of the statistic until it is found in the training image. Training images make explicit the multivariate distribution, which in most random function models is implicit [34, 37, 76, 91].

Many problems arise when inferring statistics from the data or training images. One problem is that there may not be enough data to reliably estimate multiple-point statistics. Estimating a two-point statistic like the variogram is hard enough in most cases. Stationarity is also an issue. The decision must be made to pool together data for inference. If the data do not belong to the same underlying population, the statistics extracted will not be representative of the domain under study. On the other hand, if there are not enough data to infer these statistics, the resulting simulated models will also be unreliable.

One of the main problems with multiple-point statistics is that there is a large combinatorial space to sample. Consider a template with N points and a variable that has been coded into K categories. The number of combinations whose frequencies must be found is N^K. A simple case is a template with 4 nodes and 10 classes. This results in more than a million combinations. To accurately estimate the frequencies, say up to the second decimal place, more than one hundred million replications of this pattern are required. This is one of the reasons why multiple points are not inferred from limited sample data. The use of training images alleviates this "combinatorial nightmare", but still, approximations are required to overcome the large number of replications required. Lowering the dimension of the multiple-point statistic is one approximation.

2.2.6 A Note on Positive Definiteness

Variograms and multiple-point statistics need to be positive definite. This means that during the process of kriging, the estimation variance will be positive.

Consider a covariance function such that **C** is any $n \times n$ matrix of covariances between n different sample locations. The classical definition of a positive definite matrix is that if a column vector $\mathbf{x} = (x_1, x_2, ... x_n)'$ is considered and any quadratic form $q(\mathbf{x})$ defined as:

$$q(\mathbf{x}) = \mathbf{x}'C\mathbf{x} = \sum_{i=1}^{n}\sum_{j=1}^{n} Cov_{ij} \cdot x_i \cdot x_j$$

the quadratic form is greater than 0 for all $\mathbf{x} \neq 0$ and is zero only if $\mathbf{x} = 0$ [102].

In the context of this research, the problem arises when considering integrating conventional two-point statistics (variograms or covariances) and multiple-point statistics. A covariance matrix between the single point events and the n-point events must be built. This covariance matrix will have two-point covariances, calculated to account for the relationship between pairs of single-point events, multiple-point covariances, calculated to account for the relationship between pairs of multiple-point events, and cross-covariances of single to multiple-point events, to account for pairs constituted by a single-point event and a multiple-point event.

Statistics extracted from training images are positive definite if the domain of the point to be estimated is kept constant for all the configurations of interest.

Discussion about the requirement of positive definiteness in modelling variograms can be found in [5, 23, 51]. A discussion on positive definiteness for multiple variables is presented in [126]. Finally, guidelines for licit variogram modelling can be found on [71, 75].

2.3 Geostatistical Estimation

Estimating the value of a variable at an unsampled location is done by considering the nearby information. Classical geometric methods rely on the spatial configuration of the samples used to inform the location being estimated. Polygonal, triangulation, and inverse distance weighting methods do not account for the spatial correlation between the data, that is, they do not consider the variogram as a measure of closeness and redundancy of the samples to the location of interest [82].

Geostatistical estimation techniques use the covariance or variogram. They are generically called *kriging* and are based on the minimization of the estimation variance, which is defined as the mean squared error between the estimated value and the true (unknown) value [128]. The kriging estimate is built as a linear combination of the nearby data or transformed data values and may or may not use the global mean as an additional "data". This mean does not have to be stationary. Depending on the type of kriging used, an unbiasedness condition may constrain the weights.

The most important types of kriging are briefly reviewed next. The basic equations for cokriging, that is, estimating the value of one variable given sample data from more than one source, are also presented. These are the basis for most simulation techniques that will be reviewed in **Section 2.4**.

2.3.1 Simple Kriging (SK)

Consider the residuals of Z around the mean m. Defining the new variable $Y = Z - m$, consider n data values $y(\mathbf{u}_\alpha), \alpha = 1, ..., n$. The mean of Y is zero, since they are residuals. Estimation of the Y variable at an unsampled location $\mathbf{u}_0$ is done by linearly combining these samples, that is:

$$[y(\mathbf{u}_0)]^*_{SK} = \sum_{\alpha=1}^{n} \lambda_\alpha \cdot y(\mathbf{u}_\alpha)$$

To find the weights the estimation variance is expressed as:

$$\begin{aligned}
\sigma^2_E(\mathbf{u}) &= E\left\{(Y^*(\mathbf{u}) - Y(\mathbf{u}))^2\right\} \\
&= E\left\{(Y^*(\mathbf{u}))^2\right\} - 2 \cdot E\{Y^*(\mathbf{u}) \cdot Y(\mathbf{u})\} + E\left\{(Y(\mathbf{u}))^2\right\} \\
&= \sum_{\alpha=1}^{n} \sum_{\beta=1}^{n} \lambda_\alpha \lambda_\beta E\{Y(\mathbf{u}_\beta) \cdot Y(\mathbf{u}_\alpha)\} \\
&\quad -2 \cdot \sum_{\alpha=1}^{n} \lambda_\alpha E\{Y(\mathbf{u}) \cdot Y(\mathbf{u}_\alpha)\} + E\left\{(Y(\mathbf{u}))^2\right\} \\
&= \sum_{\alpha=1}^{n} \sum_{\beta=1}^{n} \lambda_\alpha \lambda_\beta Cov\{\mathbf{u}_\beta - \mathbf{u}_\alpha\} \\
&\quad -2 \cdot \sum_{\alpha=1}^{n} \lambda_\alpha Cov\{\mathbf{u} - \mathbf{u}_\alpha\} + Cov\{0\}
\end{aligned}$$

Notice that the covariance of the Y variable is required.

To find the optimal weights, this expression is minimized by taking the partial derivatives with respect to the weights $\lambda_\alpha, \alpha = 1, ..., n$ and setting them to zero:

$$\frac{\partial[\sigma^2_E(\mathbf{u})]}{\partial \lambda_\alpha} = 2 \cdot \sum_{\beta=1}^{n} \lambda_\beta Cov(\mathbf{u}_\beta - \mathbf{u}_\alpha) - 2 \cdot Cov(\mathbf{u} - \mathbf{u}_\alpha) \qquad \alpha = 1, ..., n$$

This process yields the following system of equations known as *normal equations* or *simple kriging system* [94, 109]:

$$\sum_{\beta=1}^{n} \lambda_\beta Cov\{\mathbf{u}_\beta - \mathbf{u}_\alpha\} = Cov\{\mathbf{u} - \mathbf{u}_\alpha\} \qquad \alpha = 1, ..., n$$

In matrix notation, this system can be written as:

$$\begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} C_{10} \\ C_{20} \\ \vdots \\ C_{n0} \end{bmatrix}.$$

or

$$\mathbf{C} \cdot \lambda = \mathbf{k}$$

The variable Y can be replaced by a variable Z second order stationary, that is, whose mean and variance are constant everywhere and equal to m and σ^2, respectively.

The estimate can be re-expressed as:

$$[z(\mathbf{u}_0) - m]^*_{SK} = \sum_{\alpha=1}^{n} \lambda_\alpha \cdot (z(\mathbf{u}_\alpha) - m)$$

or

$$[z(\mathbf{u}_0)]^*_{SK} = \sum_{\alpha=1}^{n} \lambda_\alpha \cdot z(\mathbf{u}_\alpha) + \left(1 - \sum_{\alpha=1}^{n} \lambda_\alpha\right) \cdot m$$

The covariance of Y and Z is the same since the mean is equal everywhere.

2.3.2 Ordinary Kriging (OK)

If the mean is not known, a linear combination of the available data can still be utilized, however, a constraint for the weights is required to ensure unbiasedness of the estimate, that is, that the expected value of the estimate is equal to the expected value of the true values. This is simply translated into the condition that the weights sum to one. The ordinary kriging estimate and system of $n+1$ equations are:

$$[z(\mathbf{u}_0)]^*_{OK} = \sum_{\alpha=1}^{n} \lambda_\alpha \cdot (z(\mathbf{u}_\alpha))$$

$$\begin{aligned} \textstyle\sum_{\beta=1}^{n} \lambda_\beta Cov\{\mathbf{u}_\beta - \mathbf{u}_\alpha\} - \mu &= Cov\{\mathbf{u} - \mathbf{u}_\alpha\} \qquad \alpha = 1, ..., n \\ \textstyle\sum_{\alpha=1}^{n} \lambda_\alpha &= 1 \end{aligned}$$

Notice that these equations are found by minimizing the estimation variance subject to a constraint on the weights.

$$min[\sigma^2_E] \qquad s.t. \qquad \sum_{\alpha=1}^{n} \lambda_\alpha = 1$$

This is done using the Lagrange method, by adding an extra parameter which also has to be found. A new function with $n+1$ parameters is defined and its derivatives are set to be equal to 0:

$$f(\lambda_1, \lambda_2, ..., \lambda_n, \mu) = \sigma^2_E(\lambda_1, \lambda_2, ..., \lambda_n) - 2 \cdot \mu \cdot \left(\sum_{\alpha=1}^{n} \lambda_\alpha - 1\right)$$

$$\begin{aligned} \frac{\partial[f(\lambda_1, \lambda_2, ..., \lambda_n, \mu)]}{\partial \lambda_\alpha} &= 0 \qquad \alpha = 1, ..., n \\ \frac{\partial[f(\lambda_1, \lambda_2, ..., \lambda_n, \mu)]}{\partial \mu} &= 0 \end{aligned}$$

Again the variable Z is assumed to be stationary of second order, although in practice this is not strictly required. A *local stationarity* decision suffices, that is, the neighborhood of the n data used to estimate the location $\mathbf{u}_0$ does not show a clear trend [94].

2.3.3 Non-Stationary Kriging

When the mean is not constant, there are several options:

Simple Kriging with Locally Varying Mean (SK with LVM) A simple way to handle non stationarity is to use a local mean in the simple kriging equation. The global mean m is replaced by a local mean $m(\mathbf{u})$ [41, 43, 71].

Ordinary Kriging (OK) One possibility is to have OK to implicitly estimate the mean from the n data in the neighborhood of $\mathbf{u}_0$. This option is robust if enough data are available [97].

Universal Kriging (UK) or Kriging with a Trend (KT) Another possibility is to specify a polynomial shape for the trend. This is called Universal Kriging (UK) or Kriging with a Trend (KT). The estimate is built as a linear combination of these polynomial functions and the residual sample values, that is, the sample values filtered from the polynomials. Weights for the polynomials and residuals are determined by solving a kriging system with $K+1$ Lagrange parameters, to account for the K polynomial functions and the unbiasedness constraint [20, 32, 43, 64, 81, 128].

Kriging with an External Drift (KED) One last option is to estimate the mean values as a linear function of a secondary variable [43, 71, 112]. The change in the local mean of the secondary variable is assumed to be linearly linked to the mean of the primary variable. At every location a new value for the mean is calculated. This map does not have to be a polynomial fitted to the data as in KT. KED constraints local means to match the model of the smooth secondary mean and also imposes an unbiasedness condition to the weights.

2.3.4 Non Linear Variants

Kriging can be done on the original variable, on its residual around some mean, or on some transform of the data. Typical transformations are:

Normal Scores The sample data are transformed into a standard normal distribution. The assumption of multi-Gaussianity permits the development of many geostatistical techniques [167, 168, 169].

Logarithmic Since many variables in Earth Sciences are positively skewed, that is, they show a long tail of high values, a lognormal transform tends to normalize the data. This is very convenient because of the tractability of the Gaussian distribution. In the early days of geostatistics, when computer resources were scarce, many calculations were made by hand and approximations were often used. The lognormal transformation had an important place and techniques such as lognormal kriging were proposed [46, 84, 137, 143, 163].

Uniform Scores The uniform scores or rank order of the sample data could be used [17, 78, 93]; however, the only algorithm where this transformation has some applicability is Probability Kriging (PK), which is a cokriging between the indicators and the standardized rank order of the data (see **Section 2.3.6**) [162].

Indicator The variable is changed into a binary variable, where the transformed indicator represents the probability of the true value to be below a threshold. These techniques are reviewed in more detail in **Section 2.3.6**.

Factors. The variable can be decomposed into factors, that is into uncorrelated elements, such that the variable can be retrieved as a linear combination of them [148]. Kriging can be applied to estimate these factors, as in Disjunctive Kriging (DK) [6, 118, 119, 141, 145].

2.3.5 Cokriging

Cokriging is a generalization of kriging with multiple variables [43, 71, 125, 128, 171]. Note that these variables could be measuring the same attribute but with different support or precision. Copper grade in diamond drillhole samples could be considered to be a different variable than copper grades obtained from blast holes. The sample size on a core recovered from the diamond drillhole is a few kilograms, while the material from where the sample is taken in the blast hole can be as much as one tonne. They represent different regionalizations, because they are measured at different supports and their sampling errors are different.

The most intuitive case, however, is the use of two or more variables that measure different attributes, such as gold and silver grade for example, but that are highly correlated. The knowledge of one variable gives information regarding the other. This information is measured by the cross-variogram.

The general expression for the cokriging estimate with residual data is:

$$Y^*_{COK}(\mathbf{u}) = \sum_{\alpha=1}^{n} \lambda_\alpha Y(\mathbf{u}_\alpha) + \sum_{p=1}^{P} \sum_{\alpha_p=1}^{n_p} \lambda^p_{\alpha_p} Y_p(\mathbf{u}_{\alpha_p})$$

where the $Y(\mathbf{u}_\alpha), \alpha = 1, ..., n$ are the sample data for the primary variable (the variable of interest); $Y_p(\mathbf{u}_{\alpha_p}), \alpha_p = 1, ..., n_p$ are the data values for the secondary variable p at location $\mathbf{u}_{\alpha_p}$. There are P secondary variables and each one has n_p sample values $(p = 1, ..., P)$ within the neighborhood of $\mathbf{u}$. The weights λ_α and $\lambda^p_{\alpha_p}$ are determined by the cokriging system of equations:

$$\begin{bmatrix} \mathbf{C}^{00} & \mathbf{C}^{01} & \cdots & \mathbf{C}^{0P} \\ \mathbf{C}^{10} & \mathbf{C}^{11} & \cdots & \mathbf{C}^{1P} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}^{P0} & \mathbf{C}^{P1} & \cdots & \mathbf{C}^{PP} \end{bmatrix} \cdot \begin{bmatrix} \lambda^0 \\ \lambda^1 \\ \vdots \\ \lambda^P \end{bmatrix} = \begin{bmatrix} \mathbf{k}^{00} \\ \mathbf{k}^{10} \\ \vdots \\ \mathbf{k}^{P0} \end{bmatrix}. \tag{2.2}$$

where the sub-matrix $\mathbf{C}^{ij}$ is the matrix of cross-covariance values between the locations of the n_i samples of variable i and the n_j samples of variable j. If $i = j$ these terms are direct covariances and the sub-matrix is necessarily square and symmetric. The vectors λ^i and $\mathbf{k}^{i0}$ correspond to the optimal weights obtained by solving this system and the cross-covariances between the locations of the n_i data of variable i and the point of interest located at $\mathbf{u}$, respectively.

Unbiasedness constraints must be added if the variables do not have a mean of zero [82]. The traditional ordinary cokriging approach considers the following unbiasedness conditions:

$$\sum_{\alpha=1}^{n} \lambda_\alpha = 1 \quad \text{and} \quad \sum_{\alpha_p=1}^{n_p} \lambda_{\alpha_p}^{p} = 0 \ \ \forall = 1, ..., P$$

This weighting scheme limits the influence of the secondary variables. The alternative standardized ordinary cokriging scheme allows the secondary variables to have more influence [72, 82]. However, the mean of the secondary variables must be reset to the one of the primary. The unbiasedness condition becomes:

$$\sum_{\alpha=1}^{n} \lambda_\alpha + \sum_{p=1}^{P} \sum_{\alpha_p=1}^{n_p} \lambda_{\alpha_p}^{p} = 1$$

Cokriging is often avoided because of the tedious inference of cross-covariances. Simplified methods have been proposed. Collocated cokriging retains only the secondary data that is located in the location where the primary variable is being estimated. The simplification comes by assuming that the cross terms are proportional to the variogram (or covariance) of the primary variable. This is only a valid assumption if the collocated secondary sample screens completely all other secondary samples. This is known as the Markov assumption [2, 48, 179].

2.3.6 Indicator-Based Estimation

Since this research is based on an indicator framework, these techniques are reviewed in more detail to explicitly state some equations and explain the terminology that will be useful to follow the later discussion.

The non-parametric formalism of indicators was introduced in 1983 by A. G. Journel [85, 86, 87, 88, 91]. Many authors have presented this approach in great detail (e.g. see [43, 71]). This method permits the direct estimation of the conditional distribution at an unsampled location, that is, its distribution of uncertainty. It permits the random variable to have different spatial continuity for high and low values.

The indicator formalism requires the data to be coded directly as probabilities. A conditional cumulative distribution function is obtained at the location being estimated. Simulation can be performed by including the previously simulated nodes into the conditioning information, and drawing from the distribution function (see **Section 2.4.3**). Several important advantages are derived from this basic idea of directly estimating the probabilities [29, 107, 162]:

1. The correlation at different thresholds can be used, that is, a different variogram model is specified for each one of the thresholds.

2. Secondary information can be coded in the same way, which gives a great flexibility to this approach, although cokriging is still needed to integrate all sources of information.

3. Change of support can be performed by any conventional technique such as affine correction, lognormal indirect correction or the use of the discrete Gaussian model [63, 82, 121]. The type of correction should be selected according to the variance correction factor. For instance, the affine correction does not

perform well for large variance reductions. The indirect lognormal correction is in general more robust. The application of the discrete Gaussian model has the disadvantage of oversmoothing secondary modes of the point distribution, although in most applications this is not a major concern and it performs correctly [63].

4. Recoverable reserves of blocks can be calculated by truncating the corrected conditional distributions at a given cutoff.

Although very flexible, the implementation of the method can be difficult:

- The coding of soft data as if they were hard data is useful, but secondary information cannot be used as primary, even though the coding is the same. A model of Coregionalization has to be used [110, 172].
- The use of data at different supports is also a difficult task, since the correlation between the variables changes at different supports [136].

Indicator Coding

The basic idea is to code the data as probability values [41] after selecting the thresholds z_k, $k = 1, ..., K$:

$$i(\mathbf{u}_\alpha; z_k) = Prob\{z(\mathbf{u}_\alpha) \leq z_k\}$$

At every data location, there is now a vector of K indicator values. If there were n data at the beginning, then there are $n \cdot K$ indicator values after coding the data. The choice of the number of thresholds is critical for good performance of this approach: too few thresholds imply a poor discretization of the conditional distributions; a large number would reduce this problem, but larger computation and inference efforts would be needed and order relations deviations are expected (see **Section 2.3.6**) [43, 68]. Goovaerts recommends between 5 and 15 thresholds [71], Deutsch suggests a number between 7 and 11 [41]. A good practice is to match thresholds with critical values of the problem under study, and distribute them uniformly through the distribution, i.e. thresholds can be chosen at regular quantiles. Alternatively, they can be set at quantiles such that equal quantities of metal fall in each class.

Hard Data Samples with negligible sample errors are called *hard data*. The coding for hard data is:

$$i(\mathbf{u}_\alpha; z_k) = \begin{cases} 1, & \text{if } z(\mathbf{u}_\alpha) \leq z_k \\ 0, & \text{otherwise} \end{cases} \quad k = 1, ..., K \tag{2.3}$$

where $z(\mathbf{u}_\alpha)$ is the value at the data location $\mathbf{u}_\alpha$. This can be interpreted as a probability:

$$i(\mathbf{u}_\alpha; z_k) = Prob\{z(\mathbf{u}_\alpha) \leq z_k\} = F_{\mathbf{u}_\alpha}(z_k)$$

Constraint Intervals The data may only tell us that the value of z is constrained by some higher and lower limits, b_α and a_α (by physical reasons, for example). The data can be coded as follows:

$$i(\mathbf{u}_\alpha; z) = \begin{cases} 1 & , \text{ if } z_k > b_\alpha \\ undefined & , \text{ if } a_\alpha < z_k \leq b_\alpha \\ 0 & , \text{ if } z_k \leq a_\alpha \end{cases} \quad k = 1, ..., K \tag{2.4}$$

where $z(\mathbf{u}_\alpha)$ is the value at the data location $\mathbf{u}_\alpha$, a_α is the known lower limit of the variable, and b_α is the known upper limit of the variable.

Soft Categorical Data Sometimes a categorical variable can be used to condition the cumulative distribution function of the primary variable [71]. For example, if the value of the z variable and the categorical variable s (e.g. rock type) are measured at n data locations $\mathbf{u}_\alpha$, then the conditional cumulative distribution of z given the secondary categorical variable s can be calculated as:

$$F^*(z_k|s_l) = \frac{1}{\sum_{\alpha=1}^{n} i_S(\mathbf{u}_\alpha; s_l)} \sum_{\alpha=1}^{n} i_Z(\mathbf{u}_\alpha; z_k) \cdot i_S(\mathbf{u}_\alpha; s_l) \qquad k = 1, ..., K$$

where $i_S(\mathbf{u}_\alpha; s_l)$ is the indicator function of the secondary variable s (equal to 1 if the category at location $\mathbf{u}_\alpha$ is s_l and 0 otherwise).

Using the secondary information, a different conditional distribution of the primary variable will be used when kriging different locations, since the categorical conditioning variable depends on location.

Soft Continuous Data Extending the idea presented for categorical data, a continuous secondary variable v can be used to condition the primary variable z. The secondary variable is "categorized" into L classes and then used as a categorical variable to condition the cumulative distribution of the primary variable. The procedure:

- Discretize the secondary variable into L classes $(v_{l-1}, v_l]$, with v_0 commonly 0.
- Code the secondary attribute as:

$$i_V(\mathbf{u}_\alpha; v_l) = \begin{cases} 1, & \text{if } v(\mathbf{u}_\alpha) \in (v_{l-1}, v_l] \\ 0, & \text{otherwise} \end{cases} \quad l = 1, ..., L \tag{2.5}$$

- Calculate the conditional cumulative distribution of z given the secondary continuous variable v:

$$F^*(z_k|v_l) = \frac{1}{\sum_{\alpha=1}^{n} i_V(\mathbf{u}_\alpha; v_l)} \sum_{\alpha=1}^{n} i_Z(\mathbf{u}_\alpha; z_k) \cdot i_V(\mathbf{u}_\alpha; v_l) \qquad k = 1, ..., K$$

  where $i_V(\mathbf{u}_\alpha; v_l)$ is the indicator function of the secondary continuous variable v as presented before.

Indicator Kriging

The distribution of uncertainty can be inferred by kriging the indicator function at every threshold. Using the coding presented in **Equations 2.3** to **2.5**, the original n data are converted into K sets of n indicator variables. Each one of those sets of data can be used to estimate the value of the indicator at an unsampled location, that is, the probability of having $z(\mathbf{u}) \leq z_k$. The indicators at different thresholds could also be used as secondary variables to perform cokriging [43, 68, 71]. Again, there is no difference between indicator cokriging and standard cokriging, except for the transformation of the variable. A short explanation of different techniques applied to indicators is presented next.

Simple Indicator Kriging To perform simple indicator kriging [152], the stationary mean of the indicator random function is required. This mean is given by the cumulative distribution function of the random function $Z(\mathbf{u})$:

$$E\{I(\mathbf{u};z)\} = Prob\{Z(\mathbf{u}) \leq z\} = F(z)$$

The stationary simple kriging estimate of the indicator at that threshold is written:

$$\begin{aligned} [i(\mathbf{u};z)]^*_{SK} &= [Prob\{Z(\mathbf{u}) \leq z|(n)\}]^*_{SK} \\ &= \textstyle\sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u};z) \cdot i(\mathbf{u}_\alpha;z) + [1 - \sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u};z)]F(z) \end{aligned} \tag{2.6}$$

where the weights $\lambda_\alpha^{SK}(\mathbf{u};z)$ are the unique solution of the simple kriging system:

$$\textstyle\sum_{\beta=1}^{n} \lambda_\beta^{SK}(\mathbf{u};z) \cdot C_I(\mathbf{u}_\beta - \mathbf{u}_\alpha;z) = C_I(\mathbf{u} - \mathbf{u}_\alpha;z) \quad \alpha = 1,...,n \tag{2.7}$$

Notice that a covariance indicator function $C_I(\mathbf{u} - \mathbf{u}_\alpha;z)$ or, assuming stationarity, $C_I(\mathbf{h};z)$, has to be inferred for each threshold in **Equation 2.7**.

Ordinary Indicator Kriging Ordinary kriging differs from simple kriging in that the mean is unknown and therefore, unbiasedness requires the sum of the weights to be constrained to be equal to one.

The ordinary indicator kriging estimate is written:

$$\begin{aligned} [i(\mathbf{u};z)]^*_{OK} &= [Prob\{Z(\mathbf{u}) \leq z|(n)\}]^*_{OK} \\ &= \textstyle\sum_{\alpha=1}^{n} \lambda_\alpha^{OK}(\mathbf{u};z) \cdot i(\mathbf{u}_\alpha;z) \end{aligned}$$

where the weights $\lambda_\alpha^{OK}(\mathbf{u};z)$ are the unique solution of the ordinary kriging system:

$$\begin{aligned} &\textstyle\sum_{\beta=1}^{n} \lambda_\beta^{OK}(\mathbf{u};z) \cdot C_I(\mathbf{u}_\beta - \mathbf{u}_\alpha;z) + \mu_{OK}(\mathbf{u};z) = C_I(\mathbf{u} - \mathbf{u}_\alpha;z) \quad \alpha = 1,...,n \\ &\textstyle\sum_{\beta=1}^{n} \lambda_\beta^{OK} = 1 \end{aligned}$$

Again, indicator covariances have to be inferred for each threshold.

Median Indicator Kriging K variogram or covariance functions must be modelled for the procedures introduced above. The inference of the variograms at extreme low or high thresholds is, in general, difficult since there are few zeros or ones, generating a noisier experimental variogram. For thresholds close to the median, where the number of zeros and ones is roughly the same, the inference of the variogram is easier. Median indicator kriging can be applied if the K indicator random functions $I(\mathbf{u}; z_k)$ are intrinsically correlated, that is, all indicator variograms and cross variograms are proportional to a common variogram model, or equivalently, all correlograms are equal. This random function model is known as the mosaic model [87]:

$$\rho_z(\mathbf{h}) = \rho_I(\mathbf{h}; z_k) = \rho_I(\mathbf{h}; z_k, z_{k'}), \qquad \forall z_k, z_{k'}$$

The single correlogram required can be estimated using the sample z correlogram or the sample indicator correlogram at the median cutoff $z_k = M$, where $F(M) = 0.5$. The advantage of using the experimental indicator correlogram is that there are no outliers.

If all indicators are defined for all data location, that is, when there is no missing data due to the use of constraint intervals as in **Equation 2.4**, then at every location to be estimated or simulated only one kriging system must be solved. The weights will not change for different cutoffs since the data configuration and variogram remain the same.

Indicator Cokriging

Indicators at different thresholds can also be used to help infer the cumulative probability at a particular threshold. The indicator $i(\mathbf{u}; z_k)$ can be estimated using the indicator values at the data locations for the same threshold *and* for different thresholds, that is, obtain the cokriging estimate with the indicators at different thresholds to get the value at a particular threshold.

This method requires the inference of $\frac{K\cdot(K+1)}{2}$ indicator variograms and cross variograms, which makes it very demanding; however, it uses all the bivariate information given by the indicator coding, improving (theoretically) the result: a lower kriging variance than the one obtained just by kriging should be expected.

Since the mean values for the K indicator variables are the cumulative distribution function values of the corresponding thresholds, simple indicator cokriging can always be performed and should give a lower estimation variance than ordinary indicator kriging.

The simple indicator cokriging estimate can be written as:

$$[i(\mathbf{u}; z_{k_0}) - p_{k_0}]^*_{SICOK} = \sum_{k=1}^{K} \sum_{\alpha=1}^{n} \lambda^{SICOK}_{\alpha,k}(\mathbf{u}_\alpha; z_k) \cdot (i(\mathbf{u}_\alpha; z_k) - p_k)$$

where k_0 corresponds to a particular threshold in $[1, K]$ and p_k is the cumulative distribution value for threshold z_k. The weights $\lambda^{SICOK}_{\alpha,k}(\mathbf{u}_\alpha; z_k)$ are obtained by solving the simple cokriging system (**Equation 2.2**). The direct and cross covariances are replaced by indicator direct and cross covariances.

The ordinary indicator cokriging estimate is written:

$$[i(\mathbf{u}; z_{k_0})]^*_{OIKCOK} = \sum_{k=1}^{K}\sum_{\alpha=1}^{n} \lambda^{OICOK}_{\alpha,k}(\mathbf{u}_\alpha; z_k)\cdot i(\mathbf{u}_\alpha; z_k)$$

As with standard cokriging, the unbiasedness condition must be imposed to constrain the sum of weights at the threshold z_{k_0} to be one, while for every secondary variable, the weights must sum to zero.

Alternatively, the ordinary indicator cokriging estimate using the standardization of the means of the secondary variables, that is, all thresholds $z_k \neq z_{k_0}$, is:

$$[i(\mathbf{u}; z_{k_0})]^*_{OIKCOK} = \sum_{k=1}^{K}\sum_{\alpha=1}^{n} \lambda^{OICOK}_{\alpha,k}(\mathbf{u}_\alpha; z_k)\cdot (i(\mathbf{u}_\alpha; z_k) - p_k + p_{k_0})$$

In this case, the sum of all the weights must be equal to one to ensure an unbiased estimator.

Practice has shown that indicator cokriging requires considerably more inference effort and brings little improvement [68, 71]. This is particularly true when the samples for the secondary variable are collocated with the samples for the primary variable, which is called the case of homotopic sampling of all the variables. This is the case with indicators, since they are defined at the same locations, unless constraint intervals are used. In the case of heterotopic sampling, that is, when primary and secondary variables are sampled at different locations, cokriging improves the result in a more significant manner.

Probability Kriging

Instead of using all indicators for all thresholds, the original data or their standardized rank ordering could be used as a secondary variable [87, 162, 169]. This requires less effort inferring variogram and cross variogram functions; the number is reduced from $\frac{K\cdot(K+1)}{2}$ to just $2K+1$.

The standardized rank ordering corresponds to the position of the data when all the data are sorted increasingly, divided by the total number of data, i.e. $p(\mathbf{u}_\alpha) = r(\mathbf{u}_\alpha)/n$, where $r(\mathbf{u}_\alpha)$ is the rank of the datum $z(\mathbf{u}_\alpha)$.

The simple probability kriging estimate is written:

$$[i(\mathbf{u}; z_k)]^*_{PK} - F(z_k) = \sum_{\alpha=1}^{n} \lambda_\alpha(\mathbf{u}; z_k)\cdot[i(\mathbf{u}_\alpha) - F(z_k)] + \sum_{\alpha=1}^{n} \nu_\alpha(\mathbf{u}; z_k)\cdot[p(\mathbf{u}_\alpha) - 0.5]$$

where $p(\mathbf{u}_\alpha) = F(z(\mathbf{u}_\alpha)) \in [0,1]$ is the standard rank ordering of the data, which has an expected value of 0.5, $F(z) = Prob\{Z(\mathbf{u}) \leq z\}$ is the stationary cumulative distribution function of Z($\mathbf{u}$). λ_α and $\nu_\alpha, \alpha = 1, ..., n$ are the simple cokriging weights of the indicator data and the standardized rankings respectively.

The ordinary probability kriging estimate could also be written, when local deviations from the global probabilities exist.

Secondary Information

Simple Indicator Kriging with Local Prior Means In simple kriging, the decision of stationarity implies that the mean of the indicator random function is independent of the location $\mathbf{u}$ being estimated. In some cases, a secondary variable may be available that gives prior information on the mean. That probability is defined as:

$$y(\mathbf{u}; z_k) = Prob\{Z(\mathbf{u}) \leq z_k | \text{ secondary information at } \mathbf{u}\} \qquad (2.8)$$

then, the simple kriging estimate (**Equation 2.6**) using these local prior means can be rewritten as follows:

$$\begin{aligned} [i(\mathbf{u}; z_k)]^*_{SIK} &= [Prob\{Z(\mathbf{u}) \leq z_k|(n)\}]^*_{SK} \\ &= \textstyle\sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u}; z_k) \cdot i(\mathbf{u}_\alpha; z_k) + [1 - \sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u}; z_k)] \cdot y(\mathbf{u}_\alpha; z_k) \end{aligned}$$

where the weights $\lambda_\alpha^{SK}(\mathbf{u}; z_k)$ are the same than those in **Equation 2.6**.

Soft Cokriging The probabilities used as prior means in **Equation 2.8** can be used as secondary data (soft data). They are interpreted as a realization of a random variable $Y(\mathbf{u}, z_k)$, correlated with $Z(\mathbf{u}, z_k)$. A cokriging estimate using only the information at the threshold being estimated can be written.

This method requires the inference of the variogram of the primary indicator data, of the secondary variable and the cross variograms between the primary indicator variables and the secondary one.

Collocated Indicator Cokriging Practitioners often try to avoid the need of modelling too many covariances and cross-covariances simultaneously through the Linear Model of Coregionalization. Moreover, when secondary data are abundant, instability of the cokriging system can occur. One possible solution is to keep only the collocated secondary data at the location where the primary variable is being estimated. However, there is still the need to infer the variogram of the primary variable and cross variogram between the hard and soft data for every threshold. The variogram model of the secondary variable is only required at $\mathbf{h}$=0, where it is given by:

$$C_Y(0; z_k) = m_Y(z_k) \cdot [1 - m_Y(z_k)]$$

where $m_Y(z_k) = E\{Y(\mathbf{u}; z_k)\}$.

So, collocated indicator cokriging fixes the possible instability of the cokriging system, without a real simplification of the variogram model. The Markov-Bayes approximation can be done in order to alleviate the inference of cross variograms, as explained in the next section.

Markov-Bayes Algorithm Assuming that the collocated secondary data will have the biggest impact when estimating the primary variable at the same location, then the cross variogram is not required. The covariance for the secondary variable and the cross covariance can be deduced from the covariance function of the primary variable. Only a calibration parameter is required to scale this covariance.

The following relations can be proven under this assumption [180]:

$$\begin{aligned} C_Y(\mathbf{h}; z_k) &= |B(z_k)| \cdot C_I(\mathbf{h}; z_k) && \mathbf{h} = 0 \\ &= B^2(z_k) \cdot C_I(\mathbf{h}; z_k) && \forall\ \mathbf{h} > 0 \\ C_{IY}(\mathbf{h}; z_k) &= B(z_k) \cdot C_I(\mathbf{h}; z_k) && \forall\ \mathbf{h} \end{aligned}$$

where the coefficients $B(z_k) \in [-1, 1]$ are defined as:

$$B(z_k) = m^{(1)}(z_k) - m^{(0)}(z_k) \quad k = 1, ..., K$$

with

$$\begin{aligned} m^{(1)}(z_k) &= E[Y(\mathbf{u}; z_k) | I(\mathbf{u}; z_k) = 1] \quad \in [0, 1] \\ m^{(0)}(z_k) &= E[Y(\mathbf{u}; z_k) | I(\mathbf{u}; z_k) = 0] \quad \in [0, 1] \end{aligned}$$

These coefficients can be estimated by:

$$\begin{aligned} \hat{m}^{(1)}(z_k) &= \frac{1}{\sum_{\alpha=1}^{n_{IY}} i(\mathbf{u}_\alpha; z_k)} \sum_{\alpha=1}^{n_{IY}} y(\mathbf{u}_\alpha; z_k) \cdot i(\mathbf{u}_\alpha; z_k) \\ \hat{m}^{(0)}(z_k) &= \frac{1}{\sum_{\alpha=1}^{n_{IY}} [1 - i(\mathbf{u}_\alpha; z_k)]} \sum_{\alpha=1}^{n_{IY}} y(\mathbf{u}_\alpha; z_k) \cdot [1 - i(\mathbf{u}_\alpha; z_k)] \end{aligned}$$

where n_{IY} is the number of locations where both hard and soft data are available.

These relations are of interest since they greatly simplify the inference necessary to utilize secondary information. These coefficients can be calculated for more than one secondary variable, allowing integration of many different sources of information.

Correcting for Order Relations Deviations

The estimated probabilities $[i(\mathbf{u}; z_k)]^*$, $k = 1, ..., K$ generated through indicator kriging must satisfy the conditions of a cumulative distribution: they have to be non-decreasing between 0 and 1 [41, 43, 71, 87].

The kriged indicator value can lie outside the interval [0,1] because the kriged estimate may be a non-convex linear combination of the conditioning data. Lack of data in some classes and differences in the variogram models from one threshold to the next are important factors to have a non-increasing function [43, 96].

The *a posteriori* forward and downward correction of the conditional cumulative distribution functions works well in general, as documented by Deutsch and Journel [43] (**Figure 2.2**). Although more difficult in its implementation, constraining the kriging system, so that it satisfies the order relations by construction is also a solution [71].

Interpolation and Extrapolation of the Conditional Cumulative Distribution Functions

Since the number of data is limited, the distribution of local uncertainty is discretized using only five to fifteen thresholds. The continuous conditional distribution at every location $\mathbf{u}$ is then represented by a set of points $[i(\mathbf{u}; z_k)]^*$ with $k = 1, ..., K$, that lie in $[0, 1]$.

It is therefore necessary to interpolate the values between thresholds, and extrapolate the values beyond the smallest and largest values [43, 71]. This decision has a large impact in the final statistics of the model being estimated or simulated, so it has to be analyzed carefully. It is commonly sufficient to interpolate linearly between the indicator values at thresholds z_{k-1} and z_k. When extrapolating the tails, a minimum and maximum possible values should be considered and the extrapolation should not be done linearly, since this would imply a uniform distribution between the minimum value and z_1, and between z_K and the maximum value, which is often unrealistic. Power and hyperbolic models are used to extrapolate

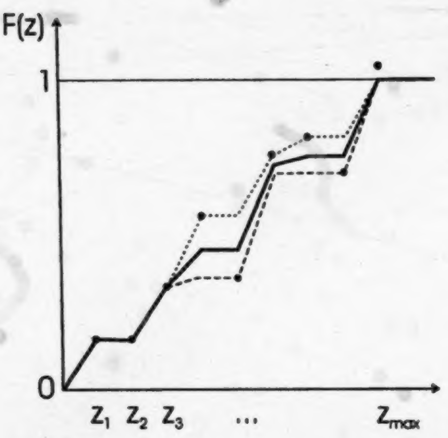


Figure 2.2: Forward and downward correction for order relation deviations. The model used is the thick line.

the distribution functions beyond the lower and higher indicator values. Another possibility is to consider the global distribution and scale it to extrapolate the tails of the distributions.

The different methods to interpolate and extrapolate are listed below:

- Linear model: Assuming a uniform distribution between the cumulative probabilities for two thresholds, or between a lower limit and the first threshold or the higher threshold and an upper limit (maximum value), the cumulative distribution function value is given by:

$$[F(z)]^*_{linear} = F^*(z_{k-1}) + \left[\frac{z - z_{k-1}}{z_k - z_{k-1}}\right] \cdot [F^*(z_k) - F^*(z_{k-1})] \qquad \forall z \in (z_{k-1}, z_k]$$

- Power model: Depending on the value of the parameter w, the power model can take a wide range of shapes (**Figure 2.3**). The cumulative distribution is calculated as:

$$[F(z)]^*_{power} = F^*(z_{k-1}) + \left[\frac{z - z_{k-1}}{z_k - z_{k-1}}\right]^w \cdot [F^*(z_k) - F^*(z_{k-1})] \qquad \forall z \in (z_{k-1}, z_k]$$

  It can be used to extrapolate the lower and upper tails of the distribution. This is done by replacing z_{k-1} and z_k by z_{min} and z_1, and using a power $w > 1$ for the lower tail, or replacing z_{k-1} and z_k by z_K and z_{max} and using a power $w < 1$ for the upper tail.

- Hyperbolic model: This model is useful to extrapolate the upper tail. As with the power model, the parameter w permits to control the shape of the function (**Figure 2.4**). The cumulative distribution is calculated as:

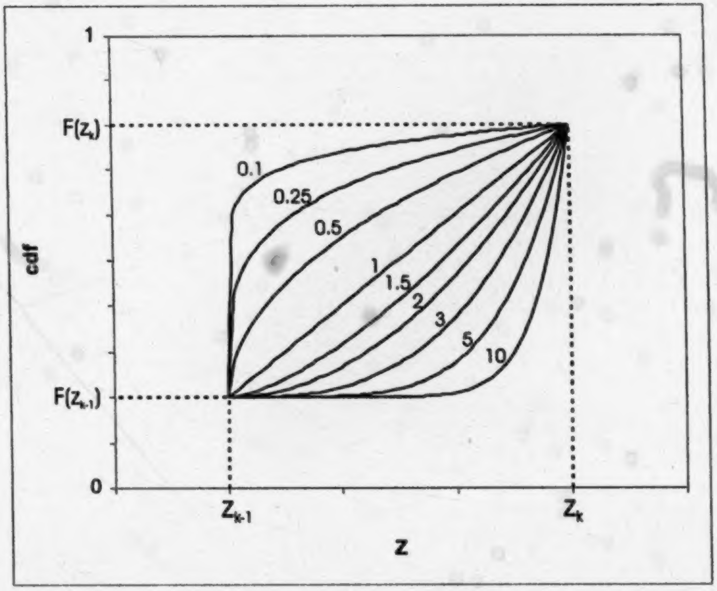


Figure 2.3: Power model for cumulative distribution function interpolation and extrapolation, given different values of the parameter w.

$$[F(z)]^*_{hyperbolic} = 1 - \frac{z_k^w \cdot [1 - F^*(z_k)]}{z^w} \qquad \forall z > z_k$$

- Re-scaling the global distribution: This can be used to extrapolate the tails of the conditional distribution. The tails of the conditional distribution will have the same shape than those of the global distribution.

2.4 Conventional Two-Point Geostatistical Simulation

Conventional geostatistical techniques exploit second-order statistics, that is, the histogram and variograms or equivalently, the covariance function. Variograms used can be direct, cross-variograms, or variograms of a transform, such as the indicator values. The common techniques are described in this section.

2.4.1 The Place of Simulation

It is known that kriging, as most estimation methods, gives a smooth map that does not look like the underlying variable (for example, see [82]). Hence, a kriged map should not be used to represent the spatial variability of the variable. Analytically it can be shown that the variance between estimates in kriging is lower than required. The difference is exactly the simple kriging variance, the so called "missing variance". When assessing uncertainty it is desired that each realization reflects the variability of the random variable.

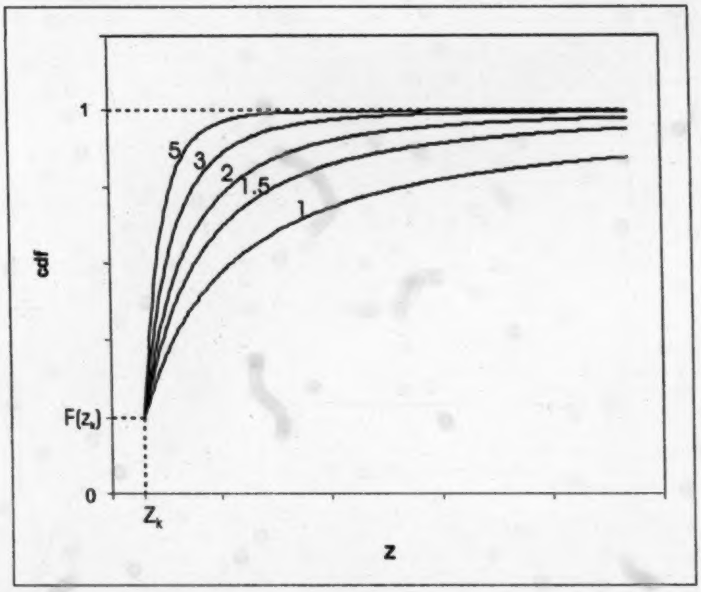


Figure 2.4: Hyperbolic model for cumulative distribution function extrapolation, given different values of the parameter w.

Simulation methods aim at reproducing the spatial variability of the underlying phenomenon. They work by drawing from a conditional distribution, that is, the missing variance is added back into the total variability of the realizations. In sequential algorithms, previously simulated nodes must be used along with the original sample data when calculating the mean of the distribution by simple kriging. This ensures the reproduction of the covariance model input in the algorithm.

Geostatistical realizations permit the calculation of *joint uncertainty*, that is, the uncertainty over larger volumes. This cannot be obtained by kriging methods. They also allow inference of response variables, such as grade above a cutoff for different supports.

Simulation does not compete with estimation, but provides a complementary result. Estimates can be obtained from multiple realizations under any measure of goodness, not only the minimization of the mean squared estimation error. Loss functions are used to come up with best estimates under different definitions of goodness. The reader are referred to [41, 88] for further details.

2.4.2 Gaussian Techniques

Although it is very rare to find a random variable whose histogram is Gaussian, methods that rely on the assumption of multivariate Gaussianity are quite common [110]. Since the original variables are not Gaussian, a transformation is required to make the sample distribution a standard normal distribution. This does not ensure that higher order moments are also Gaussian, thus one should check if the assumption contradicts the transformed data. Notice that when referring to the multivariate distribution, we are referring to it in spatial context of a single variable, and not necessarily to the case of multiple variables.

Thus, the first requirement is to have a variable that has a Gaussian histogram. This is accomplished by a simple transformation that can be performed graphically or through polynomial fitting [20, 43, 145]:

$$y(\mathbf{u}_\alpha) = G^{-1}(F_z(z(\mathbf{u}_\alpha)) = \phi\{z(\mathbf{u}_\alpha)\} \qquad \forall\alpha = 1, ..., N$$

Then, to check bivariate Gaussianity, h-scatter plots of the transformed variable could be generated. That means pairs of Y sample values that are separated by approximately **h** would need to be found. This procedure has to be repeated for different lag separation vectors **h**. Each one of these plots should show elliptical contour lines of equal probability density that are characteristic of a bivariate Gaussian distribution.

Another simple test consists of plotting the square root of the experimental variogram of normal scores over the madogram (or variogram of order 1) of normal scores, that is, half the average of absolute differences of normal scores separated by a lag distance **h**. This ratio should be constant and equal to $\sqrt{\pi}$ [53]:

$$\frac{\sqrt{\sum_{i=1}^{N(\mathbf{h})}(y(\mathbf{u}) - y(\mathbf{u}+\mathbf{h}))^2}}{\sum_{i=1}^{N(\mathbf{h})}|y(\mathbf{u}) - y(\mathbf{u}+\mathbf{h})|} = \sqrt{\pi}$$

Tests for Gaussianity at higher levels exist, however they are often inconclusive.

The reason for the popularity of Gaussian methods is that the multivariate distribution is completely defined by the knowledge of the mean and covariance function. All conditional distributions are Gaussian with the simple kriging mean and kriging variance.

Several Gaussian methods are briefly described next. They all assume the variable has been transformed to normal scores and that the transformed values do not violate the assumption of normality at higher level.

All the work is done with these normal scores. The resulting simulated values are also expressed in transformed units. A last step is then required: back-transformation of the simulated values into original units.

The results should always be checked for histogram and variogram reproduction within acceptable statistical fluctuations.

Sequential Gaussian Simulation

Sequential Gaussian simulation (SGS) is aimed at reproducing the right pattern of variability by correcting kriging. SGS works by adding a residual to the estimate. This residual is independent from the estimate obtained when kriging the normally transformed data, and it has a Gaussian distribution with mean 0, hence, it does not change the expected value of the estimate, and with variance σ_{SK}^2, giving the simulated values the right variability.

The shape of the conditional distribution is Gaussian, which ensures that the final histogram, before back-transformation, is also Gaussian.

Variogram reproduction is ensured by drawing the simulated value from a distribution with the mean and variance obtained by simple kriging [83].

Once the normal scores of the data have been obtained, the required steps for SGS are:

- Generate a random path to visit every node that has not been assigned a sample value on the grid.
- Visit each node in turn, following the random path, and perform simple kriging of the normal score transforms.
- Draw a simulated value from the Gaussian distribution with mean and variance given by the previous kriging. Notice that this is equivalent to drawing from the distribution of the random residual and adding it to the simple kriging estimated value.
- Add the simulated value to the set of hard data to be used in the subsequent kriging estimations.
- Repeat until all nodes are informed.

In practice, (transformed) sample data are often assigned to grid nodes to ensure that the samples are honored and to speed up the implementation. Only one search is required for sample data and previously simulated nodes. However, this is optional, because it may entail a significant loss of data since, if many samples are close to the same node, only the closest will be kept and used for the subsequent conditioning.

Multiple realizations can be obtained by generating a different random path and a different set of random numbers for drawing from the conditional distributions. This is implemented as a change in the seed of the pseudo-random number generator.

Matrix Simulation

Matrix or LU simulation is a very efficient algorithm when a relatively small number of nodes are to be simulated [1, 31]. It also requires transformation of the original data to normal scores. The nodes are simulated simultaneously. Matrix simulation requires decomposing a matrix of size $(n+N) \times (n+N)$, where n is the number of data values and N is the number of nodes to simulate. This decomposition generates a lower and an upper triangular matrix.

The method works by building a matrix with the covariance values between the locations with samples and nodes to be simulated. Then, the matrix is decomposed using the Cholesky method:

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}^{11} & \mathbf{C}^{12} \\ \mathbf{C}^{21} & \mathbf{C}^{22} \end{bmatrix} = \begin{bmatrix} \mathbf{L}^{11} & 0 \\ \mathbf{A}^{21} & \mathbf{L}^{22} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{U}^{11} & \mathbf{B}^{12} \\ 0 & \mathbf{U}^{22} \end{bmatrix} = \mathbf{L} \cdot \mathbf{U}$$

where $\mathbf{C}$ is the covariance matrix of size $(n+N) \times (n+N)$, $\mathbf{L}$ is a lower triangular matrix and $\mathbf{U}$ is upper triangular. The matrix $\mathbf{L}$ is decomposed into four sub-matrices, where $\mathbf{L}^{11}$ and $\mathbf{L}^{22}$ are lower triangular, $\mathbf{0}$ is a matrix of zero values and $\mathbf{A}^{21}$ is not necessarily lower triangular, to make $\mathbf{L}$ a lower triangular matrix. $\mathbf{U}$ is decomposed in a similar fashion: $\mathbf{U}^{11}$ and $\mathbf{U}^{22}$ are upper triangular sub-matrices, $\mathbf{0}$ is a matrix of zero values and $\mathbf{B}^{12}$ is not necessarily upper triangular. $\mathbf{U}$ is an upper triangular matrix. The Cholesky decomposition entails that:

$$\mathbf{L} = \mathbf{U}^{\mathbf{T}}$$

A vector $\mathbf{w}$ of size $(n+N) \times 1$ is multiplied by the the lower triangular matrix. The first n terms of the vector, denoted $\mathbf{w^1}$, are the normal scores transform of the data, and the next N, denoted $\mathbf{w^2}$, are random normal deviates. The resulting vector is the vector of normal scores of the data (first n terms) and simulated values (remaining N values).

$$\mathbf{y} = \mathbf{L} \cdot \mathbf{w} = \begin{bmatrix} \mathbf{L}^{11} & \mathbf{0} \\ \mathbf{A}^{21} & \mathbf{L}^{22} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w}^1 \\ \mathbf{w}^2 \end{bmatrix}$$

where $\mathbf{w^1} = [\mathbf{L^{11}}]^{-1} \cdot \mathbf{y^1}$, with $\mathbf{y^1}$ being the vector of normal scores of the data.

The input covariance model is reproduced:

$$\begin{aligned} E\{\mathbf{y} \cdot \mathbf{y^T}\} &= E\{\mathbf{L} \cdot \mathbf{w} \cdot \mathbf{w^T} \cdot \mathbf{L^T}\} \\ &= E\{\mathbf{L} \cdot \mathbf{w} \cdot \mathbf{w^T} \cdot \mathbf{U}\} \\ &= \mathbf{L} \cdot E\{\mathbf{w} \cdot \mathbf{w^T}\} \cdot \mathbf{L^T} \\ &= \mathbf{L} \cdot \mathbf{I} \cdot \mathbf{U} \\ &= \mathbf{C} \end{aligned}$$

A vector of random normal deviates must be generated for each realization. The generation of random normal deviates can be done easily with any random number generator and transformation of the uniform numbers into normal deviates (see **Appendix A**). Alternatively, methods for directly generating normal deviates exist [10, 113, 144].

The simulation is very fast and can be performed with high efficiency once the decomposition has been done. Improvements in the method to make it capable of handling larger grids have been developed [30].

Moving Average Methods

This method generates unconditional simulations that must be conditioned afterwards. It requires the simulated nodes to be located in a regular grid and works by calculating a weighted average of a field with known spatial covariance. The problem is to calculate the weighting function that will generate the desired covariance function.

The covariance of the initial field is often a pure nugget effect. So, the weighting function is calculated as the function that when convoluted gives the desired covariance function. The result is normal because of the Central Limit Theorem and the averaging process. It is suggested in practice to start with normal deviates to ensure that the final result will be Gaussian.

Once the normal scores have been computed, the required steps are:

- Calculate a weighting function f so that

  $$C_Y(\mathbf{h}) = f * \check{f}$$

  where $C_Y(\mathbf{h})$ is the covariance function required for the final realizations. Further details on how to obtain this function can be found in [20, 94, 110].

- Generate a field of random numbers independently drawn from a given distribution with known mean and variance (usually a Gaussian or uniform distribution are used).
- Calculate the simulated values by weighting the drawn numbers in the vicinity of the location to be simulated according to f.
- Scale the mean and variance to zero and one, to a standard normal distribution.
- Condition the simulation. This is done by adding a simulated error to the kriging estimate:

$$Y_{CS}(\mathbf{u}) = Y_K^*(\mathbf{u}) + [Y_S(\mathbf{u}) - Y_{SK}^*(\mathbf{u})] \tag{2.9}$$

  where $Y_{CS}(\mathbf{u})$ is the simulated value conditioned to the sample data; $Y_K^*(\mathbf{u})$ is the kriging estimate at that location; $Y_S(\mathbf{u})$ is the simulated value unconditional to the samples, that is, the one obtained with the moving average method in this case; and $Y_{SK}^*(\mathbf{u})$ is the kriging estimate at location $\mathbf{u}$ calculated using the simulated values at the sample locations rather than the original sample values [88].

The simulated values are then back-transformed and checked. This method is the basis for *turning bands* simulation.

Turning Bands Simulation

This method is based on the simulation of a covariance function on lines, that is, in one dimension [94]. The simulated values in two or three dimensions are obtained by averaging the projected values of the uniformly randomly distributed lines. The problem is to find the one dimensional covariance model that will generate the desired three dimensional model. Although a maximum of 15 lines can be regularly distributed in a sphere, more lines can be randomly located. This would help avoid artifacts in the method.

The steps required to apply turning bands simulation are:

- Calculate a one dimensional covariance function $C^{(1)}(\langle \mathbf{h}, \underline{\mathbf{u}} \rangle)$ so that

$$C_Y(\mathbf{h}) = \frac{1}{2\pi} \int_{\frac{1}{2}\text{unit sphere}} C^{(1)}(\langle \mathbf{h}, \underline{\mathbf{u}} \rangle) d\underline{\mathbf{u}}$$

  where $C_Y(\mathbf{h})$ is the covariance function required for the final realizations. For details on the calculation of the one dimensional covariance, see [12, 28, 94].
- Draw values with the one dimensional covariance $C^{(1)}(\langle \mathbf{h}, \underline{\mathbf{u}} \rangle)$ on L lines uniformly distributed in a unit sphere.
- Compute the three dimensional simulated values by adding the values simulated on L lines projected into the location to be simulated. Standardize the sum, dividing by $\sqrt{L}$.
- Condition the realization as in **Equation 2.9**.

The simulated values are back-transformed to original units.

Comments on Gaussian Methods

Gaussian methods are very appealing because of their simplicity. Under the multi-Gaussian assumption the shape of all conditional and marginal distributions is Gaussian, greatly simplifying inference problems. Only means and variances must be specified.

However, the Gaussian formalism assumes spatial continuity is symmetric with respect to the median and has maximum disconnectiveness at extremes, that is, if the indicator variograms of a multi-Gaussian variable are considered, they will present maximum continuity for the median threshold. Departing from it, the variograms show an increase in the nugget effect reaching, at the limit, a pure nugget effect for the highest and lowest thresholds. Variograms are symmetric, that is, the indicator variograms for say thresholds corresponding to quantiles 0.25 and 0.75 are similar. The same happens with any other pair of thresholds equidistant (in probability) to the median, for example, the 0.1 and 0.9 quantiles, the 0.2 and 0.8, etc. This concept is known in geostatistical jargon as *maximum entropy*.

This disconnectiveness may have serious consequences if the connectivity of high valued points is of importance, such as when considering flow in petroleum reservoirs. In mining this could have important consequences because of the support effect. Considering that mining is by selecting large blocks rather than "points", the connectivity of highs and lows will have an effect on the rate of change of the grade as the support gets larger. A very disconnected variable will average quickly toward the mean value, while a variable that shows connectivity of highs and lows will tend to stay high or low as the block support increases, having a slower rate of change in the grade toward the mean than in the disconnected case.

Indicator techniques are an alternative to overcome this problem. They also have other features that make them attractive.

2.4.3 Indicator Simulation

Indicator simulation avoids the need of a multi-Gaussian assumption at the bivariate level and therefore the problem of maximum entropy implicit in that assumption.

Indicator simulation uses the conditional distribution obtained through indicator kriging to draw a simulated value using Monte Carlo simulation [43, 66]. It is important to emphasize that the conditioning data used to get the conditional distributions, consist of actual data and previously simulated values within the search neighborhood. In this way, the covariance is reproduced.

The sequential simulation approach proceeds as follows:

1. Randomly pick an uninformed node.
2. Search for nearby data and previously simulated nodes.
3. Perform indicator kriging at each threshold to build the conditional distribution.
4. Draw, by Monte Carlo simulation, a value from that conditional distribution and assign it to the node.
5. Go to Step 1.

The conditioning information increases from n data to $n + N - 1$. The bigger the number of conditioning data, the bigger the kriging system. This problem is overcome by using a search neighborhood and limiting to a maximum number of data within these radii.

Variogram reproduction depends on different factors such as the size of the search neighborhood and the kriging type used.

Applications of indicator methods can be found in [92, 95] for continuous variables, and in [83] for categorical variables. A discussion about the origin of the indicator paradigm is presented in [120].

2.5 Attempts at Multiple-Point Geostatistics

Algorithms that only account for two-point statistics cannot reproduce some features that are captured by higher-order statistics. The introduction of indicator algorithms allowed different characterization of the continuity at different thresholds, which cannot be controlled by Gaussian algorithms [91]. Some novel applications of conventional simulation techniques show improvements over typical applications, by incorporating local directions of anisotropy [178] or by correcting the variogram range to account for the additional connectivity not captured by the variogram [42].

Object based methods are also used to characterize the large features first (e.g. channels) and then conventional two-point statistics are used to simulate the petrophysical variable inside the different objects [44, 62].

The direct use of multiple-point statistics in simulation has been addressed several times. The use of extended normal equations was proposed by Guardiano and Srivastava [76]. The implementation of this technique was improved by Strebelle and Journel [159], by using a search tree to find the frequencies of the multiple-point events in the training image.

Deutsch [34] applied simulated annealing for constructing reservoir models with multiple-point statistics. The difficult setting of the annealing schedule and high computational cost of this technique makes it unappealing to practitioners. Another interesting implementation was proposed by Srivastava [155] to simulate using change of support statistics, indirectly accounting for multiple-point statistics.

Another iterative technique was proposed by Caers [13] that is based on the use of neural networks to model the conditional distribution function in a non-linear fashion.

All implementations proposed assume that multiple-point statistics are available. They consider training images for their inference. The reproduction of features that belong to the training image but not to the underlying process that is being simulated has not been addressed properly. We may want to reproduce the general appearance of the training image but not all its details. Caers [13] uses a technique to avoid overtraining the neural network, however the question of which features should be extracted from the training image is not answered. Furthermore, transferring statistics from the training image to the realization is a problem. The univariate and bivariate statistics of the training image may not be exactly the same as those of the phenomenon. Once again, the use of multiple-point statistics inferred from the data does not have this problem, since the statistics are consistent to a common spatial law.

A quick review of the methods currently available to simulate incorporating multiple-point statistics is presented next.

2.5.1 Object-Based Methods

Object based techniques are a natural way to model geological shapes. In reservoir characterization, the petrophysical properties change drastically from one facies to another. The facies are generally located in a certain depositional environment. For instance, sand is easier to be found in a channel than outside it. Therefore the generation of the surfaces or volumes that define the objects is seen as first-order heterogeneity in the reservoir characterization. Inside these objects, second and third-order heterogeneities can be modelled [44].

Although this approach is appealing since it mimics the genesis of geological formations, it is difficult to implement because of the large number of parameters required to stochastically generate the objects. Inference of these parameters is generally done from similar reservoirs. The other problem is that conditioning to data is not easy. The objects must be moved to match the data and several iterations are required. Active research is being done in this field [170].

2.5.2 Variogram-Based Techniques

Other techniques have been proposed to account for long range connectivity. Xu and Journel [178] proposed an approach based on simulating the local angle of anisotropy, in order to reproduce curvilinear features of the true underlying phenomenon. They then applied a conventional two-point technique to simulate the petrophysical property, using the previously simulated angles of anisotropy.

Another approach consists on increasing the range on the variogram model to account for the longer range connectivity, which is not captured directly by the variogram. Deutsch and Gringarten [42] used an annealing approach to accomplish this task.

The use of isofactorial models to characterize bivariate behavior in a framework similar to that of disjunctive kriging has been proposed to generate numerical models with the same two-point statistics, but with control over the connectivity of extremes [52].

2.5.3 N-Point Connectivity Function

Gaussian methods suffer from the maximum entropy of extremes [120]. This means that, when considering the continuity of extreme high or low values, it tends to a pure nugget effect.

Journel and Alabert [91] used the n-point connectivity function to show the improvement in the reproduction of long range connectivity of realizations constructed with the sequential multiple indicator algorithm, versus realization built with Gaussian techniques.

They defined multiple steps connectivity based on the n-point function:

$$\phi(n) = E\left\{\prod_{j=1}^{n} I[\mathbf{u} + (j-1)\mathbf{h}; z]\right\}$$

where n is the number of connected points considered at the same time and z is a threshold.

They did not use this statistics in the simulation but showed the better performance of indicator techniques compared with Gaussian simulation.

Deutsch [34] incorporated this statistics in numerical models using simulated annealing for petroleum reservoir characterization. This is the only practical application of this concept.

The n-point connectivity function is a non-centered multiple-point indicator covariance function, similar to a run. As shown in **Section 3.2**, if the random function was known, the n-point connectivity function could be calculated before-hand, without requiring a realization. In practice, this is possible only in two cases: (1) in the multi-Gaussian case, where the multivariate distribution is fully defined by its mean and covariance function; each conditional probability is retrieved by simple indicator kriging, and (2) the independent case (pure nugget effect) is easily computed:

$$\phi(n) = \prod_{j=1}^{n} F(z) = [p_z]^n$$

Although indicator simulation improves the result when the desired multivariate distribution departs from the multi-Gaussian case, this high-order continuity cannot be captured only by two-point statistics. These higher-order features will considerably change the resulting uncertainty after the transfer function.

2.5.4 Extended Normal Equations

Guardiano and Srivastava [76] introduced the generalization of the indicator algorithm and use of the extended normal equations (see also [83]). Conventional indicator kriging is just an approximation of the more general theory presented here.

The conditional expectation of an indicator variable can be calculated exactly by considering a linear combination of the following events:

- The indicator values at the same threshold $i(\mathbf{u}_\alpha; z_{k_0}), \alpha \in (n)$,
- The indicator values at different thresholds $i(\mathbf{u}_\alpha; z_k), \alpha \in (n), k = 1, ..., K, k \neq k_0$, and
- All products of indicators from all thresholds, that is, products of two indicator values at a time, three indicator values and so on.

The conditional expectation can be written as a function of these events:

$$
\begin{aligned}
F(\mathbf{u}; z_{k_0}|(n)) &= P(Z(\mathbf{u}) \le z_{k_0}|(n)) \\
&= E\{I(\mathbf{u}; z_{k_0})|(n)\} \\
&= \phi\Big\{\{i(\mathbf{u}_\alpha; z_k), \alpha \in (n), k = 1, ..., K\}, \\
&\quad \{i(\mathbf{u}_\alpha; z_k) \cdot i(\mathbf{u}_\beta; z_{k'}), \alpha \in (n), \beta \in (n), \\
&\quad k = 1, ..., K, k' = 1, ..., K\}, \\
&\quad \ldots \\
&\quad \{\prod_{k=1}^{K} \prod_{\alpha \in (n)} i(\mathbf{u}_\alpha; z_k)\}\Big\}
\end{aligned}
$$

where z_{k_0} is one of the k thresholds considered.

Inferring the weights to linearly combine these events calls for the knowledge of the entire spatial law of the variable, which is never possible. Simplifications are then required.

First, the conditional probability is approximated by dropping all the terms that involve indicators at different thresholds. Only the indicators at the same threshold z_{k_0} are used. All cross-correlation between indicators at different thresholds and products of indicators, also called multiple point indicators, is ignored. This first approximation is done not because the inference of the cross-correlations is too difficult, but because, in general, the improvement in the resulting simulation does not justify the increase work required. This is particularly true because the covariances between the different events must be positive definite to ensure that the system has a solution and that this solution is unique.

The conditional expectation can be written as a function of the conditioning information in the following manner:

$$
\begin{aligned}
&E\{I(\mathbf{u}; z_{k_0})|I(\mathbf{u}_\alpha; z_{k_0}) = i(\mathbf{u}_\alpha; z_{k_0}),\ \alpha \in (n)\} = \phi'\{i(\mathbf{u}_\alpha; z_{k_0}), \alpha \in (n)\} \\
&\quad = a_0 + \sum_{\alpha \in (n)} a_1(\alpha) \cdot i(\mathbf{u}_\alpha; z_{k_0}) \\
&\quad + \sum_{\alpha \in (n)} \sum_{\alpha' \in (n), \alpha \neq \alpha'} a_2(\alpha, \alpha') \cdot i(\mathbf{u}_\alpha; z_{k_0}) \cdot i(\mathbf{u}_{\alpha'}; z_{k_0}) + \ldots \\
&\quad + a_n \cdot \prod_{\alpha \in (n)} i(\mathbf{u}_\alpha; z_{k_0})
\end{aligned}
$$

Notice that the function ϕ is different than ϕ'. The indicators and products of indicators at the same threshold are now being used to estimate the conditional expectation.

The 2^n coefficients a_0, $a_1(\alpha)$, $a_2(\alpha, \alpha')$, ..., a_n in the last expression correspond to the extended indicator kriging weights and can be determined by an extended system of 2^n normal equations [76, 109].

Using a training image to extract the multiple-point covariances ensures the positive definiteness condition on the covariance matrix.

The implementation of this technique was improved by Strebelle and Journel [159], by using a search tree to find the frequencies of the multiple-point events in the training image. The methodology as applied to reservoir modelling is outlined in [160].

The classical application of indicators considers only the use of univariate and bivariate statistics (the sample histogram and the covariance or variogram function), since the positive definite modelling of higher order covariances is difficult. The first (n+1) terms of the previous expansion are retained for kriging and only two-point covariances are used, that is, the standard covariance as defined in **Equation 2.1**.

2.5.5 Simulated Annealing

Simulated annealing (SA) is a general optimization algorithm that is capable of incorporating as many statistics as required to the simulation process [8, 20, 34, 54, 60, 103, 147]. The algorithm will honor all of the statistics if they are consistent with each other and the optimization parameters are set correctly. The basic idea is to start with a realization that does not honor the statistics and perturb the nodes until the statistics are close enough to the target. This is done by defining an objective function that corresponds to a weighted sum of component objective functions. Each one of these components corresponds to a measure of mismatch between the target statistics and the current statistics, which are expressed as a mathematical expression.

In general, the objective function is written:

$$O = \sum_{i=1}^{N_c} \omega_i O_i$$

where N_c is the number of components in the objective function, ω_i are the weights assigned to each one of the components, and O_i is the mismatch value for component i.

For example, this function could be composed by the mismatch in histogram reproduction, defined as the difference in the cumulative frequencies measured at some quantiles for the model being simulated versus the target histogram, and a mismatch in variogram reproduction, composed by differences between the target variogram model and the variogram calculated from the realization being perturbed, for a number of lag distances. In this case:

$$O = \omega_1 \cdot \sum_{i=1}^{Q} \left[F_Z^{model}(q_i) - F_Z^{target}(q_i)\right]^2 + \omega_2 \cdot \sum_{i=1}^{n_{lag}} \left[\gamma^{model}(\mathbf{h}_i) - \gamma^{target}(\mathbf{h}_i)\right]^2$$

where $F_Z^{model}(q_i)$ and $F_Z^{target}(q_i)$ correspond to the cumulative frequency for a given quantile q_i for the model being perturbed and for the reference statistics, Q is the number of quantiles in which the cumulative frequencies (interval [0,1]) has been discretized, $\gamma^{model}(\mathbf{h}_i)$ and $\gamma^{target}(\mathbf{h}_i)$ are the variogram values for a lag $\mathbf{h}_i$ in a specific direction, and n_{lag} is the number of lags considered.

SA allows to incorporate in the same manner, variograms in multiple directions, indicator variograms, multiple-point histograms and any other statistics or constraint such as conditioning data [34].

A key characteristic of SA is that some bad changes are accepted, that is, even if the perturbation increases the value of the objective function, it may be kept.

The rule for accepting or rejecting a change is based on the Gibbs or Boltzmann probability distribution, which gives the name to the algorithm, since it was used to model the energy of molecules in the physical process of annealing [41].

The fact that some bad changes are conditionally accepted differentiates SA from most optimization algorithms, where all bad changes are rejected. The probability of acceptance, given by the Boltzmann distribution is:

$$P\{accept\} = \begin{cases} 1 & \text{if } O_{new} \leq O_{old} \\ e^{\frac{O_{old}-O_{new}}{t}} & otherwise \end{cases}$$

where t is a parameter equivalent to the product of the Boltzmann constant k_b and the temperature T in the application to the physical process. By analogy, t is called the temperature in SA; O_{old} and O_{new} are the values of the objective function before and after the perturbation, equivalent to the difference in Gibbs free energy ΔE in the physical process of annealing. All good changes and some bad changes are accepted. As in the physical process of annealing, the temperature decreases with time letting the molecules to reorganize in a state of lower energy. In SA, the temperature must be lowered as the algorithm runs. In the numerical implementation, the number of perturbations attempted is associated with time.

There are many consideration before attempting to run a SA algorithm:

Initial Realization The initial realization is in general spatially random with the target histogram, because otherwise, long range features may be difficult to undo, taking longer for the algorithm to converge to the desired statistics.

Objective Function The components of the objective function will dictate which features will be present in the simulated model. These components should not be inconsistent with each other, otherwise the model will not reach a low objective function value because of the incompatibility of requirements. The components must make physical sense. Furthermore, the objective function must be designed so that if all statistics are matched, it equals zero.

Stop Criteria The obvious criterion is to stop if the objective function is very close to zero. That means that the statistics of the simulated model are very close to the target ones. Deutsch [41] suggests a value of 1% of the initial value of the objective function. A second criterion for stopping is CPU time. If the algorithm does not converge within reasonable time, it should be stopped. If the objective function was still decreasing that means that the problem is too complex and may require more perturbations, hence a longer CPU time to converge. If the objective function has converged to a value not close to zero, this means that the components of the objective function may be conflicting with each other or that the decision rule, also called *annealing schedule* was not set up properly (see next). The formulation of the problem must be revised in this first case and the parameters of the annealing schedule, revised, in the second case.

Perturbation Mechanism As mentioned, swapping of nodes randomly selected as a perturbation mechanism will preserve the histogram. An alternative is to randomly select one node and draw a new value from the global target distribution. It has also been proposed to draw from a conditional distribution built by indicator kriging the surrounding nodes in a given template [41], or by calibration with a secondary variable [45].

Updating of Objective Function The re-calculation of the objective function can be done by updating the initially calculated objective function with the changes due to the modification of the node (or nodes) perturbed. This makes the algorithm much more efficient in terms of CPU time than re-calculating the entire objective function every time, as illustrated by Deutsch in [41, 43, 45]. A typical example is updating the variogram after one node has been perturbed. The new value of the variogram for a given lag is calculated as:

$$\begin{aligned}\gamma_{new}^{model}(\mathbf{h}_i) &= \gamma_{old}^{model}(\mathbf{h}_i) + \\ &\frac{1}{2 \cdot N(\mathbf{h}_i)}\left[(z^{new}(\mathbf{u}) - z(\mathbf{u}+\mathbf{h}_i))^2 - (z^{old}(\mathbf{u}) - z(\mathbf{u}+\mathbf{h}_i))^2\right]\end{aligned}$$

where $\mathbf{h}_i$ is a particular lag distance, $N(\mathbf{h}_i)$ is the number of pairs encountered in the model to calculate the variogram at that lag, $z^{new}(\mathbf{u})$ and $z^{old}(\mathbf{u})$ are the values of the node being perturbed after and before the perturbation has taken effect, and $z(\mathbf{u}+\mathbf{h}_i)$ is a node $\mathbf{h}_i$ apart from the node perturbed. This fast updating of the objective function speeds up the simulation process considerably, since the number of operations required is greatly reduced.

Annealing Schedule The annealing schedule refers to the parameters that specify how the temperature is reduced. The temperature parameter t must be lowered to allow convergence. However, convergence is not guaranteed and depends on how this parameter is changed during the simulation. As the temperature decreases, bad changes will have a lower probability of being accepted, that is, the realization will tend to stay in the same state (nodes will not change) unless the changes are favorable. In practice the temperature is lowered with six control parameters (see [41] for more details):

- The initial temperature t_0 should be set to a high value.
- The reduction factor $\lambda \in (0, 1)$ is a multiplicative factor to reduce the temperature if a maximum number of attempted perturbations K_{max} is reached at the same temperature, or if a maximum number of accepted perturbations is reached at that temperature.
- The simulation will stop if K_{max} is reached S times, that is, if the number of perturbations accepted at a given temperature has not been reached in the last S attempted temperatures.
- The tolerance in the objective function to define convergence ΔO, corresponds to about 1% of the initial value of the objective function.

Further discussion on SA for geostatistcal applications in petroleum can be found in the book by Deutsch [41]. Deutsch also [34] applied simulated annealing for integration of production data in petroleum reservoir modelling. Casar-González end Suro-Pérez [18] applied this method to simulate vuggy formations in a Mexican offshore field.

Multiple-point statistics can be input as easily as the histogram or variogram in the algorithm. Deutsch [34] proposed the use of multiple-point histograms in an annealing framework. The idea is to define a pattern of p points and code the data as indicators. Then 2^p possible combinations of zeros and ones are possible. Each one of these is coded with a number. For example for a pattern of four points:

$$Index_{MP}(i_1, i_2, i_3, i_4) = 1 + \sum_{j=1}^{4} 2^{j-1} \cdot i_j$$

where i_1, i_2, i_3, and i_4 are the indicator transforms for the four data points. In this case, a histogram of frequencies of the sixteen combinations can be used as a target statistics and the multiple point histogram can be added to the objective function. Connectivity functions can also be specified in the same way.

Srivastava proposed honoring multiple-point statistics by controlling histograms at different supports [155].

Multiple-points histograms have also been used more recently in simulation in an annealing framework by Qiu and Kelkar [139].

Although very flexible, annealing suffers of one drawback. The fine tuning of the parameters is often difficult, and convergence for problems with complex objective functions, for example, with multiple components and many statistics to honor, may be too slow for practical applications. However, as computers get faster and more powerful, this technique will surely have a place in solving difficult problems in the future.

2.5.6 Iterative Methods and Markov Chain Monte Carlo Methods

Iterative methods are based on the work by Metropolis et al. [122]. They have been used in spatial statistics [144] and were introduced in geostatistics by Srivastava [154].

The idea behind iterative techniques is to start with a realization and perturb the nodes until the model converges to the desired statistics. However, the perturbation mechanism differs from annealing. It uses a Markov Chain to go from one state to the next, that is, there is a transition matrix that defines the probability of going from one state (one arrangement of values on the nodes) to another. The problem is to find a transition matrix that satisfy some properties that ensure that, after a large number of perturbations, the simulation will converge to the desired state. Several methods exist: the Metropolis algorithm, the Barker's algorithm, and the Metropolis-Hastings algorithm [14, 20]. A simpler application corresponds to the Gibbs sampler [20, 43]. In this case, only one node is perturbed at a time, by drawing from a conditional distribution built from the surrounding information. This conditional distribution can be built by gathering information from several sources, such as indicator variograms, and potentially multiple-point statistics such as connectivity functions, and multiple-point histograms. The nodes in the grid are visited many times in random order and at every location the current value is discarded and a new one is drawn from its current local distribution.

Caers [13] presented another technique based on the use of neural networks to model the conditional distribution function in a non-linear fashion. This method requires a training image to train the neural network. Coefficients are calculated that allow the inference of the local distributions of uncertainty given surrounding data. Although convergence of this method is ensured, the speed cannot be calculated beforehand. Applications to facies modeling can be found in [16, 153].

2.6 Discussion

Geostatistics provides a way to generate numerical models of regionalized variables that help in decision making. Conventional techniques exploit up to second-order statistics. Higher-order statistics are defined by the random function model, that is often multivariate Gaussian. The incorporation of multiple-point statistics derived from actual data into the modelling process should lead to models that better represent the reality.

Although many methods have been proposed to consider multiple-point information, they have not been widely applied for several reasons:

1. Inference of multiple-point statistics is not an easy task and depending on the type of application, they may require a modelling step to ensure a consistent mathematical formulation of the problem. Regression type algorithms and the solution of linear systems of equations are particularly sensitive to inconsistencies in the input statistics, generating unreliable results.

2. Considering a training image representative of the phenomenon under study means it is assumed that the training image and the sample data belong to the same stationary population. There is a need for the training image to share the cumulative distribution and exactly represent the multivariate distribution desired for the modelled phenomenon.

3. Setting of the parameters for object-based methods, simulated annealing and iterative methods is a difficult task. In the first case, many parameters about the shape, dimensions, and orientation of the objects have to be set. For SA, the setting of the annealing schedule requires a deep understanding of the method and previous experience. As for Markov Chain Monte Carlo methods the choice of the algorithm and complex theory behind it make it unappealing.

The proposed work aims at investigating some of these issues and proposing new solutions.

Chapter 3

Incorporating Multiple-Point Runs in Geostatistical Simulation

This chapter covers the development and implementation of a simulation algorithm that accounts for runs in multiple directions. This algorithm considers simulating single-point events using multiple-point data. This is done in an hierarchical indicator framework, that is, the runs are simulated for one threshold at a time. Once the simulation is completed for the first threshold (either the highest or lowest cutoff value), the generated runs are used as conditioning information for the next threshold. The implementation of this algorithm shows artifacts that invalidate the practical use of the proposed method, however the research is considered valuable due to the insight it provided.

Some general concepts are recalled. Key concepts and the definition of runs are presented in **Section 3.1**.

The analytical derivation of the frequency of runs for a general case is exposed, with application to the multi-Gaussian and independent cases (**Section 3.2**).

Hierarchical indicator simulation accounting directly for multiple-point runs is presented in **Section 3.3**. Implementation details are presented. Some examples in one and two dimensions are presented and the problems encountered are discussed.

Finally, some comments about trying to directly account for runs in simulation and discuss an alternative approach to incorporate multiple-point statistics are presented (**Section 3.4**).

3.1 Introduction

3.1.1 Key Concepts

Multiple-Point Statistics as Runs Multiple-point (MP) statistics are different summaries of the multivariate spatial distribution of the variable. These statistics are estimated by the frequency of occurrence of the spatial arrangement of the multiple points (see related discussion on **Section 2.2**). In general, if the variable of interest is continuous, the indicator paradigm can be used to characterize it as a binary event at several thresholds (see **Figure 3.1**).

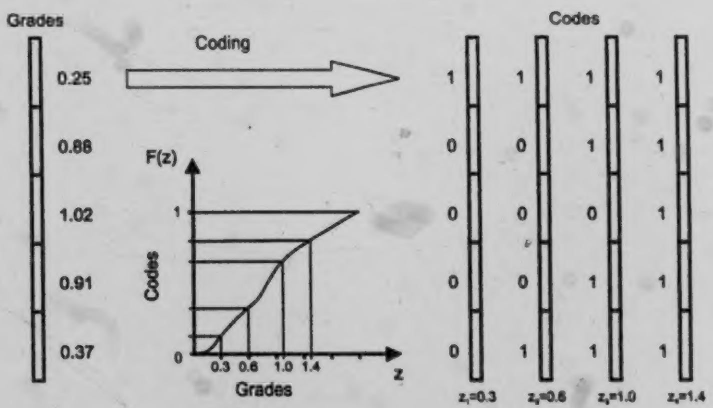


Figure 3.1: Coding a continuous variable into indicators for different thresholds.

As discussed in **Section 2.3.6**, the data is considered for a fixed threshold as being above or below it. Multiple thresholds can be considered, allowing a discretization of the range of variability of the data. Accounting for multiple locations adds additional complexity to the problem. The inference of multiple-point statistics is done by calculating the frequency of the arrangement of indicators for the multiple locations from some source. This can be a training image, as in most proposed methods, or data with some repeated pattern, such as drillholes, as in the approach proposed in this research. The larger the number of points considered in this multiple-point arrangement, the harder to infer the statistic from data, since a larger number of samples is required as the number of elements of the MP pattern increases. For instance, there are 32 possible combinations of five locations that take a binary outcome.

In general multiple-point events can be considered for any number of points, but in practice, only a few points, say 3 to 6, are sufficient to improve the appearance and performance of numerical models. Deutsch [34] showed how a simple 4 point pattern generates realizations that look much more realistic even reproducing long range features. Most application of multiple-point statistics utilize a small number of points in their definition of the pattern (see for example [15, 153, 159, 160]).

In this chapter, we focus on utilizing multiple-point configurations arranged in lines. Furthermore, the points must be equidistant (**Figure 3.2**). Other configurations could also be used if enough replications of the spatial arrangement are available. As in most practical applications, these constrains on the multiple-point configuration can be relaxed and the multiple-point statistics can be inferred from data approximately equidistant or that approximately falls in some patterns.

One advantage of using linear strings of data is that inference is possible from drillhole data. A drawback is that curvilinear features will not be captured by this statistic. In any case, resorting to training images is always an alternative

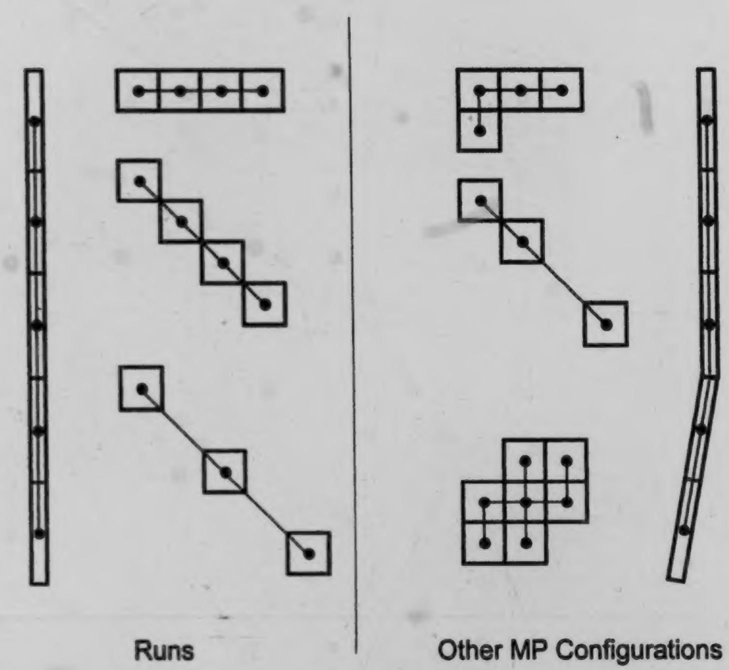


Figure 3.2: Multiple-point configurations valid as runs are shown on the left. The configurations on the right could also be used if enough replications were available for inference.

to make inference possible.

Nesting of Runs As illustrated in **Figure 1.1**, nesting of runs naturally occurs for continuous variables, when the indicator coding is used. Runs above the current threshold are also above any lower threshold. This property can be used to simulate hierarchically. The runs at one threshold condition the next threshold (see below).

Hierarchical Simulation The idea of simulating one indicator at a time, starting with the highest (or the lowest) and proceeding hierarchically to the lower (higher) ones is attractive because order relation deviations would be avoided by construction. Furthermore, the reproduction of the proportions within each class would be controlled at every threshold.

Order relation deviations are one of the main problems of indicator simulation. Each indicator is estimated separately, so the algorithm does not impose the conditions required for a cumulative distribution. Although indicator cokriging (see **Section 2.3.6**) would partially solve this problem of consistency between thresholds, it could not be applied in a hierarchical framework and reproduction of runs would be difficult.

This hierarchical approach has been used before for indicator simulation. Further discussion can be found on [26] and it is further investigated in **Appendix D**. This is a key concept for the method proposed (see **Section 3.3**).

3.1.2 The Theory of Runs

The results presented here are based on a paper by A. M. Mood published in 1940 [123]. Mood summarized most of the work done previously by other authors and can be considered as the basis for the majority of the subsequent statistical studies on runs (see for example [57, 67, 73, 124, 149, 161]). Mood derived the "*distribution of runs of given length both from random arrangements of fixed numbers of elements of two or more kinds, and from binomial and multinomial populations*". He also gives the limiting form of these distributions as the sample size increases. Those limit distributions are all normal. The results are based on combinatorial analysis, so independence between the elements is assumed at all times. Binary events are of particular interest to us, since these results are to be applied in the indicator framework.

Let us first consider a sequence of uniform random numbers between a and b. A threshold z_k can be set and then rename each number with a zero, if it is greater than z_k, or a one, if it is less than or equal to z_k. That is, the values are coded as indicators (see **Equation 2.3**). Since the numbers are uniformly distributed, they can be considered as drawn from a Bernoulli distribution with probability of drawing a one equal to $p = \frac{z_k - a}{b-a}$. Zeros are drawn with probability $q = 1 - p$. Now that a sequence of zeros and ones is available, the length of strings of ones and zeros can be evaluated. This is what is called runs.

For uniform random sequences, the distribution of runs of given lengths is known, so this property can be used to test pseudo-random number generators (see **Appendix A** for further examples). The special case when $z_k = 0.5$, $a = 0.0$, and $b = 1.0$ originated the so called tests of runs above and below the mean (or the median).

The following example shows how to calculate the runs for a sequence of uniform random numbers between 0 and 1, using the median as a threshold ($z_k = 0.5$).

Let the sequence be:

0.35, 0.56, 0.12, 0.11, 0.84, 0.76, 0.77, 0.45, 0.61, 0.51

This sequence would generate the following sequence of zeros and ones (values above or below the median):

1, 0, 1, 1, 0, 0, 0, 1, 0, 0

and the sequence of lengths of runs above/below the median would be:

1, 1, 2, 3, 1, 2

Considering that there are n_0 zeros and n_1 ones (and knowing that $n = n_0 + n_1$) then the proportions $q = \frac{n_0}{n}$ and $p = \frac{n_1}{n}$ of values above and below the threshold can be calculated. The total number r of runs above and below z_k should follow a normal distribution with the following mean and variance [123]:

$$m_r = E\{r\} = 2 \cdot n \cdot p \cdot q$$

$$\sigma_r^2 = E\left\{(r - E\{r\})^2\right\} = 4 \cdot n \cdot p \cdot q \cdot (1 - 3 \cdot p \cdot q)$$

When the threshold is the median (or the mean) of a uniform distribution then, the parameters are simply:

$$m_r = \tfrac{n}{2} \qquad \sigma_r^2 = \tfrac{n}{4}$$

The number of runs above z_k of length i, noted $m_{r_{0i}}$, can be calculated as:

$$m_{r_{0i}} = \frac{(n_1+1)^{(2)} n_0^{(i)}}{n^{(i+1)}}$$

where the factorial $x^{(a)}$ corresponds to $x^{(a)} = x \cdot (x-1) \cdot (x-2) \cdot \ldots \cdot (x-a+1)$

The number of runs above z_k of length greater than or equal to i, noted $m_{s_{0i}}$, can also be calculated:

$$m_{s_{0i}} = \frac{(n_1+1) n_0^{(i)}}{n^{(i)}}$$

The covariance and variance between the number of runs of zeros of different lengths i and j are given by:

$$\begin{aligned}
\sigma_{ij} &= \frac{n_1^{(2)}(n_1+1)^{(2)} n_0^{(i+j)}}{n^{(i+j+2)}} - \frac{n_1^2 (n_1+1)^2 n_0^{(i)} n_0^{(j)}}{n^{(i+1)} n^{(j+1)}} \\
\sigma_{ii} &= \frac{n_1^{(2)}(n_1+1)^{(2)} n_0^{(2i)}}{n^{(2i+2)}} + \frac{(n_1+1)^{(2)} n_0^{(i)}}{n^{(i+1)}} \left(1 - \frac{(n_1+1)^{(2)} n_0^{(i)}}{n^{(i+1)}}\right)
\end{aligned}$$

The expected value for the total number of runs of zeros and its variance are given by:

$$\begin{aligned}
m_{r_0} = E\{r_0\} &= \frac{(n_1+1) n_0}{n} \\
\sigma_{r_0}^2 &= \frac{(n_1+1)^{(2)} n_0^{(2)}}{n n^{(2)}}
\end{aligned}$$

Finally, when n_0 and n_1 are fixed, the distribution of the total number of runs of elements of zeros (or ones) is asymptotically normal:

$$r_0 \sim N\left(\frac{n_0 n_1}{n}, \frac{n_0^2 n_1^2}{n^3}\right)$$

When the number of elements are random variables drawn from a binomial population, then the numbers n_0 and n_1 are not fixed and the results change. The mean and variance of runs of zeros of length i, and the covariance between runs of zeros of lengths i and j, become:

$$\begin{aligned}
m_{r_{0i}} &= q^i p\{(n-i-1)p+2\} \\
\sigma_{ij} &= q^{i+j} p^2 \{(n-i-j)^{(2)} p^2 + (n-i-j)p(1+5q) \\
&\quad +6q^2 - ((n-i-1)p+2)((n-j-1)p+2)\} \\
\sigma_{ii} &= q^{2i} p^2 \{(n-2i)^{(2)} p^2 + (n-2i)p(1+5q) \\
&\quad +6q^2 - ((n-i-1)p+2)^2\} + q^i p((n-i-1)p+2)
\end{aligned}$$

where p and $q = 1 - p$ are the probabilities of drawing a one and a zero respectively.

And the limit distribution of the total number of runs is asymptotically normal with the following mean and variance:

$$r \sim N\,(2npq\ ,\ 4npq(1 - 3pq))$$

Most of the moments of the distribution of runs for a random uniform case can be predicted. Analytical or approximate expressions for correlated sequences are of interest (in particular, the multi-Gaussian case, see **Section 3.2**).

For geostatistical applications, the indicator coding (**Equation 2.3**) can be applied to define two types of elements (zeros and ones). The continuous random variable Z is transformed into a binary random variable I.

Consider the following sequence of uniform numbers in [0,1] and the three sequences coded for thresholds 0.25, 0.50, and 0.75.

z-value:	0.35	0.07	0.85	0.94	0.66	0.48	0.65	0.35	0.79	0.19
$i(z; z_1 = 0.25)$:	0	1	0	0	0	0	0	0	0	1
$i(z; z_2 = 0.50)$:	1	1	0	0	0	1	0	1	0	1
$i(z; z_3 = 0.75)$:	1	1	0	0	1	1	1	1	0	1

A run of length L above a threshold z_k can be identified as a sequence of $L + 2$ adjacent nodes valued as zeros, except for the first and last nodes, valued as ones. The first run above the threshold 0.25 in the previous example is of length $L = 1$. The second run has a length $L = 7$ and is highlighted below.

z-value:	0.35	0.07	**0.85**	**0.94**	**0.66**	**0.48**	**0.65**	**0.35**	**0.79**	0.19
$i(z; z_1 = 0.25)$:	0	1	0	0	0	0	0	0	0	1

Notice that the runs above the threshold are nested, that is, runs above a threshold contain the runs above a higher threshold. This is shown next, where runs above the thresholds 0.25, 0.50, and 0.75 are highlighted, showing the nesting.

z-value:	0.35	0.07	0.85	0.94	0.66	0.48	0.65	0.35	0.79	0.19
$i(z; z_1 = 0.25)$:	0	1	0	0	0	0	0	0	0	1
$i(z; z_2 = 0.50)$:	1	1	0	0	0	1	0	1	0	1
$i(z; z_3 = 0.75)$:	1	1	0	0	1	1	1	1	0	1

The same happens with runs below a threshold.

z-value:	0.35	0.07	0.85	0.94	0.66	0.48	0.65	0.35	0.79	0.19
$i(z; z_1 = 0.25)$:	0	1	0	0	0	0	0	0	0	1
$i(z; z_2 = 0.50)$:	1	1	0	0	0	1	0	1	0	1
$i(z; z_3 = 0.75)$:	1	1	0	0	1	1	1	1	0	1

As seen in this example, indicator coding facilitates the application of the concept of runs in geostatistical simulation. Several additional exploratory examples using runs are presented in **Appendix B**.

3.2 Analytical Derivation of the Frequency of Runs

In general, multiple-point events can be analytically derived if the multivariate spatial distribution is known. This is very uncommon. We show the multivariate Gaussian and independent cases.

3.2.1 General Case

A run of length L of elements below a threshold can be seen as the event of having a string of nodes of length $L+2$, such that the first and the last values are above the threshold z_k and all other nodes are below the threshold value. The separation distance between nodes is $\mathbf{h}$.

This multiple-point event occurs with the following joint probability:

$$\begin{aligned} Prob\{\text{Run of length L}\} \;&= \\ & Prob\{Z(\mathbf{u}) > z_k, Z(\mathbf{u}+\mathbf{h}) \leq z_k, ..., Z(\mathbf{u}+L\cdot\mathbf{h}) \leq z_k, Z(\mathbf{u}+(L+1)\cdot\mathbf{h}) > z_k\} \end{aligned}$$

This joint probability can be calculated by a recursive application of Bayes' postulate, that is, if **A** and **B** are two events (multiple-point events or not), the probability of both events happening is equal to the probability of the first event conditional to the second multiplied by the probability of the second event occurring:

$$Prob\{\mathbf{A},\mathbf{B}\} = Prob\{\mathbf{A}|\mathbf{B}\} \cdot Prob\{\mathbf{B}\} \tag{3.1}$$

In the case of runs, the multiple-point event "run of length L" has a probability of occurring given by:

$$\begin{aligned} Prob\{\text{Run of length L}\} \;&= \\ & Prob\{Z(\mathbf{u}) > z_k | Z(\mathbf{u}+\mathbf{h}) \leq z_k, ..., Z(\mathbf{u}+L\cdot\mathbf{h}) \leq z_k, Z(\mathbf{u}+(L+1)\cdot\mathbf{h}) > z_k\} \cdot ... \cdot \\ & Prob\{Z(\mathbf{u}+L\cdot\mathbf{h}) \leq z_k | Z(\mathbf{u}+(L+1)\cdot\mathbf{h}) > z_k\} \cdot \\ & Prob\{Z(\mathbf{u}+(L+1)\cdot\mathbf{h}) > z_k\} \end{aligned}$$

3.2.2 The Multi-Gaussian Case

The conditional probabilities involved in the calculation can only be completely retrieved in a case where the spatial law is fully known. In the multi-Gaussian case all conditional distributions are characterized by the mean vector and covariance matrix.

Let us denote the multi-Gaussian variable Y. It could be the normal score transform of Z. Code the data as indicators at threshold y_k and rewrite the expression for the joint probability as:

$$\begin{aligned} Prob\{\text{Run of length L}\} \;&= \\ & Prob\{I(\mathbf{u}) = 0 | I(\mathbf{u}+\mathbf{h}) = 1, ..., I(\mathbf{u}+L\cdot\mathbf{h}) = 1, I(\mathbf{u}+(L+1)\cdot\mathbf{h}) = 0\} \cdot \\ & Prob\{I(\mathbf{u}+\mathbf{h}) = 1 | I(\mathbf{u}+2\cdot\mathbf{h}) = 1, ..., I(\mathbf{u}+L\cdot\mathbf{h}) = 1, I(\mathbf{u}+(L+1)\cdot\mathbf{h}) = 0\} \cdot ... \cdot \\ & Prob\{I(\mathbf{u}+l\cdot\mathbf{h}) = 1 | I(\mathbf{u}+(L+1)\cdot\mathbf{h}) = 0\} \cdot Prob\{I(\mathbf{u}+(L+1)\cdot\mathbf{h}) = 0\} \end{aligned}$$

Now, conditional probabilities can be calculated in the multi-Gaussian case, by simple indicator kriging [91]:

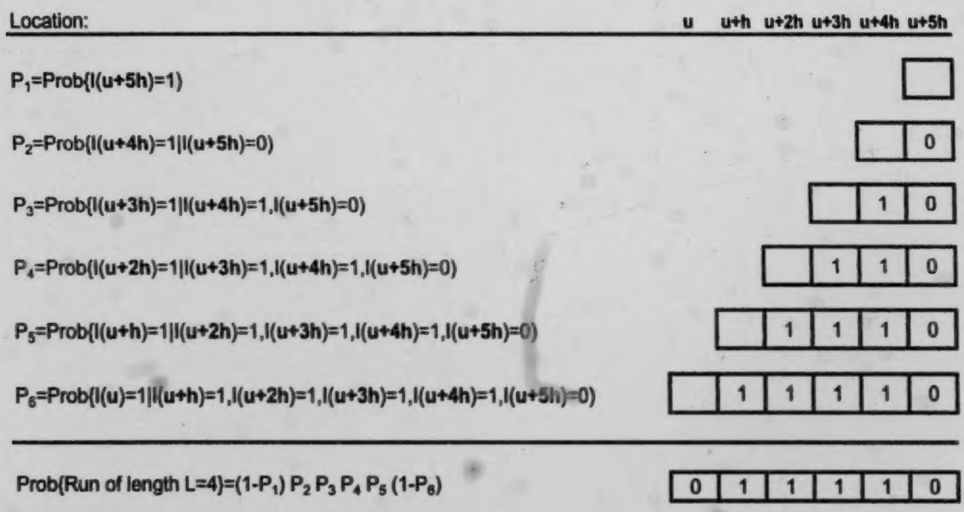

Location:
u
u+h
u+2h
u+3h
u+4h
u+5h
P1=Prob{I(u+5h)=1}
P2=Prob{I(u+4h)=1|I(u+5h)=0}
P3=Prob{I(u+3h)=1|I(u+4h)=1,I(u+5h)=0}
P4=Prob{I(u+2h)=1|I(u+3h)=1,I(u+4h)=1,I(u+5h)=0}
P5=Prob{I(u+h)=1|I(u+2h)=1,I(u+3h)=1,I(u+4h)=1,I(u+5h)=0}
P6=Prob{I(u)=1|I(u+h)=1,I(u+2h)=1,I(u+3h)=1,I(u+4h)=1,I(u+5h)=0}
Prob(Run of length L=4)=(1-P1) P2 P3 P4 P5 (1-P6)
0 1 1 1 1 0

Figure 3.3: Illustration of the increasing conditioning in the calculation of the joint probability of having a run of length L. L=4 in this example. The calculation of the terms $P_1, P_2, ..., P_6$ is done by simple indicator kriging.

$$Prob\{Y(\mathbf{u}) \leq y_k|(n)\} = Prob\{I(\mathbf{u}) = 1|(n)\} = \sum_{j=1}^{n} \lambda_j \cdot I(\mathbf{u}_j) + (1 - \sum_{j=1}^{n} \lambda_j) \cdot F(y_k)$$

where the $\lambda_j, j = 1, ..., n$ are the solution of the system:

$$\sum_{j=1}^{n} \lambda_j \cdot C_I(\mathbf{u}_k, \mathbf{u}_j) = C_I(\mathbf{u}_k, \mathbf{u}_i) \quad k = 1, ..., n$$

The increasing conditioning is illustrated in **Figure 3.3**. Notice that the covariance function $C_I(\mathbf{u}_k, \mathbf{u}_j)$ has to be calculated from the continuous covariance of the Gaussian variable, as shown in [43].

3.2.3 Example

The previous results can be tested by constructing a realization of a multi-Gaussian variable with a known covariance function. Then, the continuous simulated values can be coded as indicators for a given threshold, say the median, and the indicator variogram can be calculated for that threshold [88]. This experimental indicator variogram must be modelled or derived analytically from the known covariance $C_Y(\mathbf{h})$. Simple indicator kriging is used to calculate the conditional probabilities with increasing conditioning, as illustrated in **Figure 3.3**. A chart showing a comparison between the theoretical calculation and the experimental result is presented in **Figure 3.4**. An almost perfect match between the two suggests that the n-point connectivity curve (see **Section 2.5.3**) calculated from a realization obtained through a multi-Gaussian simulation algorithm can be predicted analytically. Journel and Alabert [91] used the n-point connectivity function to illustrate the improvement

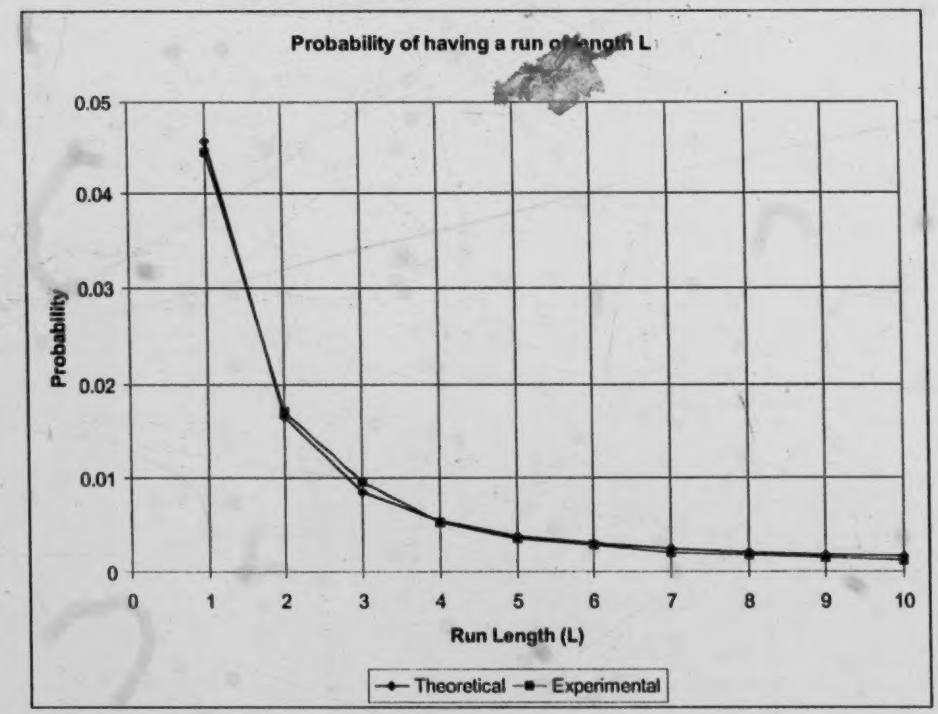


Figure 3.4: Comparison of theoretical and experimental results for the calculation of the probability of having a run of length L. This is also called the n-point connectivity function.

obtained by using indicator simulation over Gaussian simulation. They calculated the n-point connectivity function from the realization obtained via a Gaussian technique. As stated above, this calculation could have been done without recursing to a numerical rendition, since it is analytically defined.

3.2.4 The Random Case: Relation with Mood's Results

Mood's results [123] are calculated in a finite domain of size n. The probabilities derived using a recursive application of Bayes' postulate (**Equation 3.1**) are defined in an infinite (ergodic) domain. When all elements are drawn independently, then Bayes postulate becomes:

$$Prob\{\mathbf{A},\mathbf{B}\} = Prob\{\mathbf{A}\} \cdot Prob\{\mathbf{B}\}$$

Thus, all conditional distributions are reduced to their marginals. The probability of having a run of elements below a threshold of length L is simply:

$$
\begin{aligned}
Prob\{\text{Run of length L}\} &= Prob\{I(\mathbf{u})=0\}\cdot Prob\{I(\mathbf{u}+\mathbf{h})=1\}\cdot \ldots \cdot \\
&\quad Prob\{I(\mathbf{u}+L\cdot\mathbf{h})=1\}\cdot Prob\{I(\mathbf{u}+(L+1))=0\} \\
&= q\cdot p\cdot p\cdot \ldots \cdot p\cdot q \\
&= p^L\cdot q^2
\end{aligned}
$$

Taking Mood's expected values for runs of length L, when the elements above and below the threshold are drawn from a binomial population, and considering the total number of events of length $L+2$ in a sequence of n nodes, the proportion of multiple-point events of size $L+2$ of which the nodes have a run configuration can be calculated. Thus, the corresponding probability can be derived by taking the limit as n approaches infinity:

$$
\begin{aligned}
Prob(r_{1L}) &= \lim_{n\to\infty}\frac{p^L q\{(n-L-1)q+2\}}{n-(L+2)+1} \\
&= \lim_{n\to\infty}\left(p^L q^2+\frac{2p^L q}{n-L-1}\right) \\
&= p^L\cdot q^2
\end{aligned}
$$

Thus, the result derived by Mood is exactly the same as the analytical derivation.

The total number of runs can be calculated as the infinite sum of the expected number of runs of length L:

$$
\begin{aligned}
m_r &= m_{r_0}+m_{r_1}=E\left\{\sum_{L=1}^{\infty}(r_{1L}+r_{2L})\right\} \\
&= \sum_{L=1}^{\infty} n\left(Prob(r_{1L})+Prob(r_{2L})\right) \\
&= \sum_{L=1}^{\infty} n\left(p^L q^2+q^L p^2\right) \\
&= \sum_{L=1}^{\infty} n\left(p^L(1-p)^2+q^L(1-q)^2\right)
\end{aligned}
$$

Rearranging the values and using the following result for the infinite sum [80]:

$$
\lim_{n\to\infty}\sum_{L=1}^{n} x^L=\frac{x}{1-x}\qquad |x|<1
$$

we get:

$$
\begin{aligned}
m_r=E\{r\} &= n(p(1-p)+q(1-q)) \\
&= n(p\cdot q+q\cdot p) \\
&= 2npq
\end{aligned}
$$

which is exactly the value given by Mood.

3.2.5 Discussion

The results presented above show the simplicity and ease of calculation when the multivariate distribution is known. Real data do not show in general a multi-Gaussian or pure random behavior, hence the analytical derivation and modelling of the multiple-point runs is not straightforward. Inference must be done from some source of information that characterizes the multivariate spatial distribution of the variable. Borrowing these features from a training image is the easiest option, since in that case, the spatial law is assumed to be represented in this reference image. As pointed out earlier, it has the advantage of being exhaustive and consistent (positive definite). A second option is to infer these multiple-point statistics from data. Unfortunately many problems arise when extracting statistics from samples:

1. We can find strings of data from drillholes. Channel samples are also generally taken in lines. Blastholes are generally drilled in pseudo-regular arrays. Two dimensional configurations could be obtained from blasthole samples. Approximations could be made to find runs in the horizontal plane, in several directions. Reconciling these one and two dimensional data with a three dimensional consistent model may be difficult (see next).

2. Ensuring positive definiteness of statistics from different sources is difficult. The combination of one and two dimensional information must lead to a valid three dimensional mathematical model of the correlation of multiple-points. This problem is also encountered when considering training images.

3. Kriging methods or more generally, the normal equations, require a positive-definite model for the covariances of all orders, which is particularly difficult when considering cross-covariances between statistics of different order. However, some updating techniques permit to avoid the modelling of these covariances. Disregarding the fact that the mathematical model must be valid may allow calculations, but will be reflected in the final result as order relations deviations or other artifacts, which are a sign of inconsistency of the model.

The advantage of working with statistics extracted from actual data is a data-driven model. The more statistics can be reliably inferred from the data, the more information this model will share with the data set used. This should reflect favorably in the performance of the numerical models built.

3.3 Hierarchical Indicator Simulation

A first attempt to directly simulate in an indicator framework, accounting for multiple-point runs, is documented. The algorithm considers a hierarchical approach to erode a field. Runs above and below several thresholds are considered in different directions. The general idea is described next; the input information required, selection criterion, implementation details and examples are also included.

3.3.1 Methodology

The general methodology is presented next. Every point is discussed more extensively at the end of the list.

1. Start at the highest threshold z_K.
2. Fill the grid with zeros except at conditioning points, where the z-values are coded as indicators. A value of zero means the node has a z-value above z_K.
3. Calculate the current histograms of runs above and below the threshold for all the directions of interest.
4. Visit every node in the grid that is not a conditioning value or has not been frozen at a previous threshold (none of them has been frozen yet at z_K). Nodes that at the current threshold have already been switched to one should also be skipped during this process.
    - Switch its value to one.
    - Recalculate the histogram of runs considering this change.
    - Calculate the value of the selection function with the current values of the nodes in the grid. Save that value for comparison.
    - Reset the value of the node to zero to restore the original state.
5. Assign a value of one to the node with highest selection function value.
6. Update the histograms of runs above and below the threshold in all directions.
7. Go back to 4 and repeat until enough nodes have been switched to honor the proportion below the threshold.
8. Once enough nodes have a value of one, draw a simulated value between the current threshold and the adjacent higher threshold (or a maximum value if the current threshold is the highest) at all locations that have a value of zero, which means their z-value is higher than the threshold value.
9. Move to the adjacent lower threshold and reset all the nodes that do not have a z-value assigned to zero. Recall that only conditioning data and nodes with a value higher than the previous higher threshold have already a z-value assigned.
10. Go back to 3 and repeat all the steps at the current threshold.
11. Once the lowest threshold is reached and the switching of nodes is completed, then all nodes with a value of zero are assigned a z-value simulated between the lowest threshold z_1 and the adjacent higher threshold z_2. The nodes with a value of one are below the lowest threshold and a simulated value is drawn between a minimum value z_{min} and the lowest threshold z_1. This concludes the simulation.

The algorithm can be seen as an erosion process. The grid starts filled with very high values and is then eroded. The erosion is performed using some decision rule to select the nodes. This idea is repeated in smaller and smaller domains as lower thresholds are considered. If a node has been set to be above the current threshold, that means it is also above all subsequent lower thresholds, hence it is frozen and not considered in the domain for the lower thresholds. The domain for the first threshold is the entire area minus the nodes that correspond to conditioning samples. The nodes not eroded for that threshold are frozen and cannot be moved for the next thresholds.

The algorithm was initially implemented to reproduce the indicator variogram. Conclusions for this case are illustrated in **Appendix D**. The discussion presented here is for the case of reproducing runs.

The algorithm is illustrated on **Figure 3.5**. Three thresholds have been considered and a one-dimensional grid of ten nodes is being simulated. Starting at the highest threshold z_3, all ten nodes are simulated to be above or below that threshold. The runs above and below that threshold should be reproduced (see discussion on the selection of nodes, next). Nodes that have been assigned a zero are higher than z_3, thus higher than z_2 and z_1 as well. Simulated values are drawn between z_3 and a maximum value z_{max}, and with some extrapolation shape. All the nodes that have a value higher than z_3 are not available at the following threshold z_2, they are discarded from the domain for simulating at z_2. The nodes in the reduced domain are now simulated, considering the frozen nodes as conditioning data. The algorithm continues in this fashion up to the last threshold z_1. The nodes assigned a zero in the binary simulation are now drawn between z_1 and z_2 with some interpolation shape, and the values that are below z_1 are drawn as well between a minimum value z_{min} and z_1 (see **Section 2.3.6**). This concludes the simulation.

Input Information

The algorithm requires two basic input statistics:

- Proportion of nodes below the threshold, that is, the cumulative distribution function value of the threshold.

- Histograms of runs above and below the threshold. They can be inferred from drillhole data or from any source where the samples represent the same support and are linearly located.

Runs are calculated as illustrated in **Figure 3.6** [131], that is, taking what classically is called a run of length L, we can consider it as one run of length L plus two runs of length $L-1$, plus three runs of length $L-2$, and so forth. A run of length L corresponds to i runs of length $L-i+1$, with $i = 1, ..., L$. Consider a simplified case with a run of length 3 and two runs of length 2. The histogram of accumulated runs would be calculated as the sum of the three histograms, see **Figure 3.7**. The reason for changing the classical definition of runs to the one presented in **Figure 3.6** is that the construction of histograms of runs with the new definition generates a plot with a decreasing number of runs as the length increases. This is done to further control long runs, since a long run (in the sense of the classical definition) will have components for all shorter lengths.

Inference of these statistics requires data in form of drillholes or wells, and a representative distribution to obtain the proportions for each class.

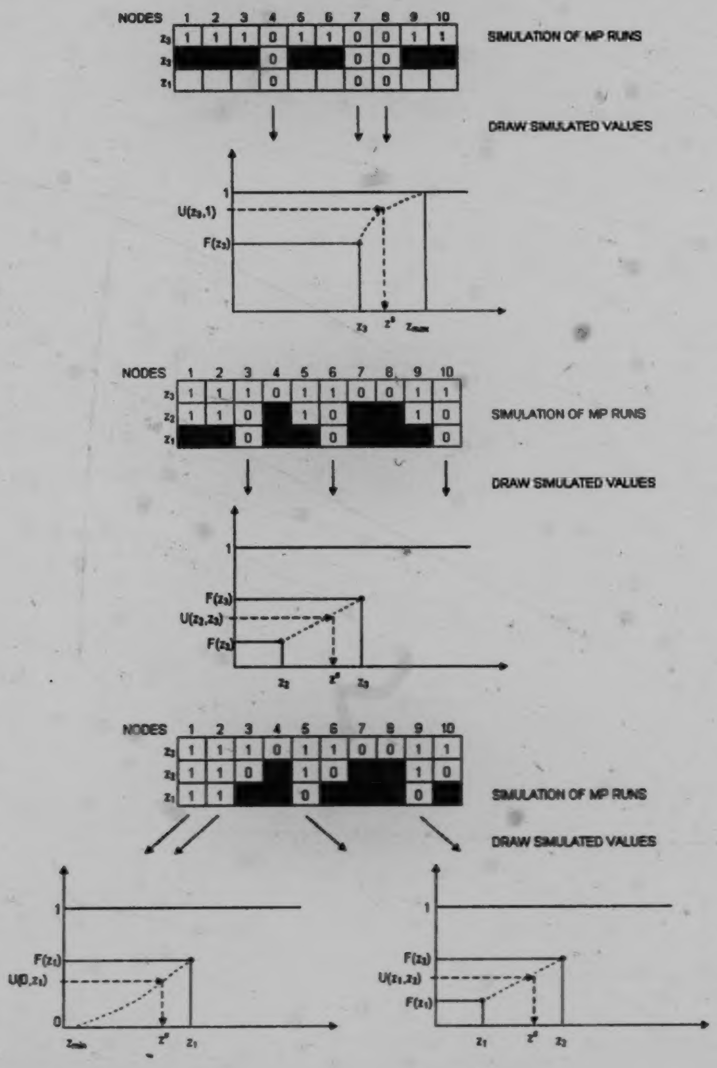

NODES
1 2 3 4 5 6 7 8 9 10
z3
z2
z1
SIMULATION OF MP RUNS
DRAW SIMULATED VALUES
1
U(z3,1)
F(z3)
z3
z*
zmax
U(z2,z3)
F(z2)
z2
F(z1)
U(0,z1)
0
zmin
z1
U(z1,z2)

Figure 3.5: Schematic of hierarchical indicator simulation of runs. Top: all nodes are available to be simulated. Nodes that have been assigned a zero are frozen for all subsequent thresholds, since they are higher than z_3. Simulated values are drawn between z_3 and a maximum value z_{max}. The remaining nodes are considered for the next threshold. Middle: the nodes not frozen are simulated at threshold z_2. The frozen nodes from the previous threshold are used to condition the simulation of the remaining ones. Again, nodes with indicator values equal to zero are discarded from the simulation domain for the subsequent threshold z_1, while the ones valued as one become the domain for the simulation at the following threshold. Bottom: finally, the last threshold z_1 is simulated with a reduced domain. After nodes have been assigned an indicator value, simulated z-values are drawn between thresholds, and between the lower threshold z_1 and a minimum value z_{min}.

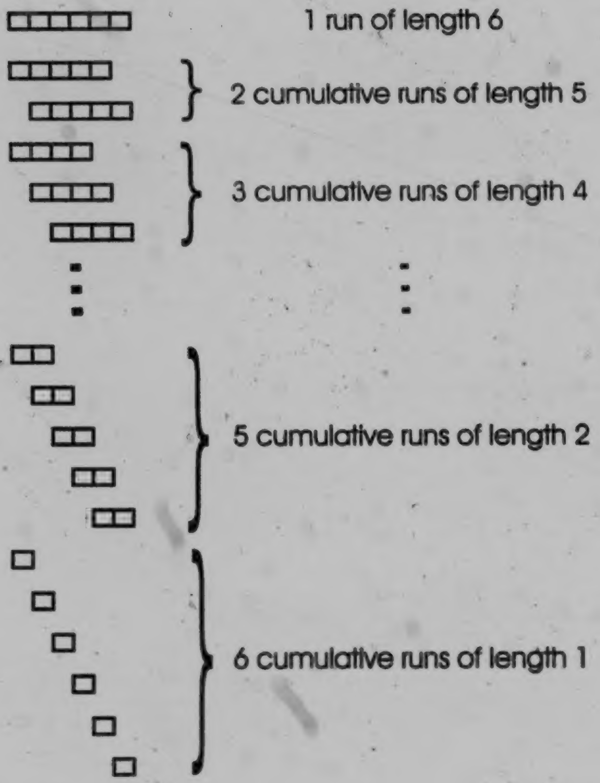

1 run of length 6
2 cumulative runs of length 5
3 cumulative runs of length 4
5 cumulative runs of length 2
6 cumulative runs of length 1

Figure 3.6: The concept of "accumulated runs": one single run of length 6 corresponds to 2 accumulated runs of length 5, 3 accumulated runs of length 4,..., and 6 accumulated runs of length 1.

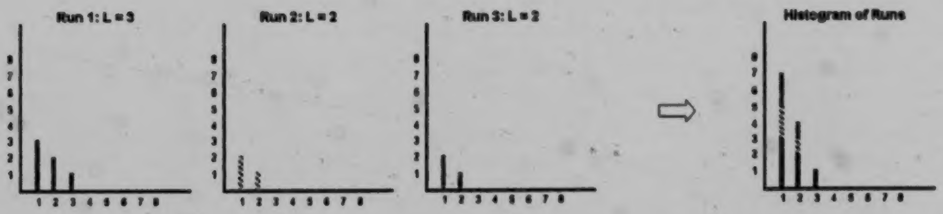


Figure 3.7: The histogram of accumulated runs given three runs of length 3, 2, and 2.

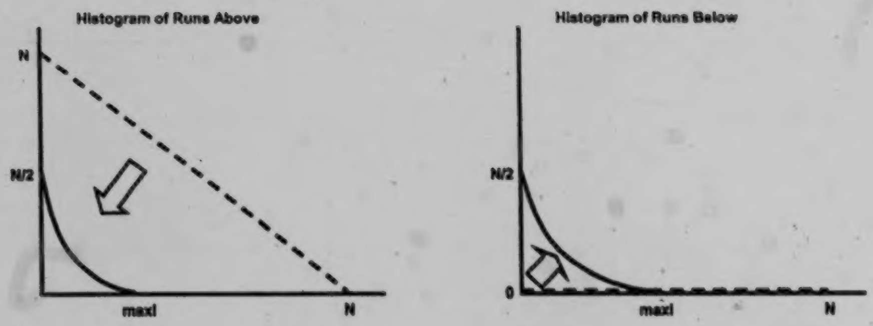


Figure 3.8: Histograms of runs above and below the threshold at the beginning of the simulation (dashed lines). The current histograms must converge to the targets (solid lines).

Selection of Nodes

The selection of which node to switch is critical to obtain the desired result. The selection is based on a function that quantifies the closeness of the current histograms of runs to the target histograms of runs for that particular threshold.

Recall that the histogram of runs to be honored is defined for several directions. The current histogram is calculated by counting how many runs of zeros (above the threshold) and ones (below the threshold) are found in every direction of interest. The starting histograms in a case without conditioning data are shown in **Figure 3.8.**

Runs are counted up to a maximum length. The mismatch for every length, between the current state of the simulated field and the target values, is calculated for runs above and below the threshold for all direction of interest.

A decision rule is applied to pick a node whose indicator value is going to be switched to one. Only nodes that have not been frozen are considered in this operation. Hence, conditioning data are not included in this process. The node to be switched is selected to bring the current histograms of runs as close together as possible to the target histograms. The selection criterion could hence be considered "greedy" in optimization terms, that is the fastest path to convergence is followed.

The measure of closeness of the current and target histograms is a weighted function of mismatches for runs of different lengths, giving equal value to all directions. More complex rules could have been considered, however simplicity was preferred in order to extract some insight from sensitivity analysis of the result as a function of changes in this selection rule. The histogram of runs above and below the threshold are also equally weighted, that is, none of them is favored.

The decision of which one to switch is made based on a selection function that will penalize changes that do not significantly improve the matching between the current and desired histograms of accumulated runs above and below the threshold. This is considered for all directions where the histograms of runs are available.

The selection process considers evaluating at all available nodes at the threshold the value of a selection function:

$$s(\mathbf{u}_i) = \prod_{l=1}^{2 \cdot maxl} f_{above}(l) \cdot f_{below}(l), \quad i = 1, ..., N \tag{3.2}$$

where l is the length of a cumulative runs, $maxl$ is the maximum length of runs in the target histograms, N is the total number of nodes in the grid, and

$$f_{above}(l) = \begin{cases} 0.05 + 0.95 \cdot e^{-(\frac{-\Delta}{a})^w} & , \text{ if } \Delta < 0 \\ 0.05 + 0.95 \cdot e^{-(\frac{\Delta}{a})^w} & , \text{ otherwise} \end{cases}$$

with Δ being the difference between the number (frequency) of cumulative runs of length l in the current histogram, noted $freq_{curr}(l)$, and the number of cumulative runs of the same length in the target histogram, noted $freq_{targ}(l)$:

$$\Delta = freq_{curr}(l) - freq_{targ}(l)$$

$f_{below}(l)$ is defined exactly as $f_{above}(l)$ but with

$$\Delta = freq_{targ}(l) - freq_{curr}(l)$$

The parameters a (scaling) and w (power) give the shape of the function $f(l)$, as illustrated in **Figure 3.9**.

The idea behind the selection function (**Equation 3.2**) is to find a path towards convergence of the simulated runs to the desired runs. It was found, by experimenting with the selection function, that the simulation will tend to take easy paths, such as building up some runs and ending up with very long runs, if the selection function did not force the best possible improvement in histogram matching. Although highly heuristic, the selection function tends to generate realizations with histograms of runs close to the target.

Tuning was always necessary to ensure good reproduction of the target statistics. As the complexity of the problem increases, that is, as more directions, thresholds and longer runs are accounted for, the tuning becomes more difficult. Parameters were then found by trial and error.

Consider the starting case with only one node below the threshold (coded as a one) and all other nodes above it (coded as zeros). There are two possibilities: we can either cut the long run of zeros into two of approximately the same size, or we can switch the node at the end of the run of zeros, shortening the long run by one node (see **Figure 3.10**) and increasing the length of the run of ones, of length one to two. The first option brings both histograms closer to the target than the second option. Anything in between these two extreme cases will yield an improvement in the matching of histograms of runs that is not as good as the first case, or as bad as the second case. The selection function must favor nodes as in the first option. This will bring the histogram of runs of the simulated model closer to the target ones faster.

The selection function presented in (**Equation 3.2**) reflects this concept by giving a value close to one to changes that favor the closeness of the model and target histograms. If the mismatch is exactly zero for all lengths of runs considered, the selection value will be one and the node will be picked, since it has the highest

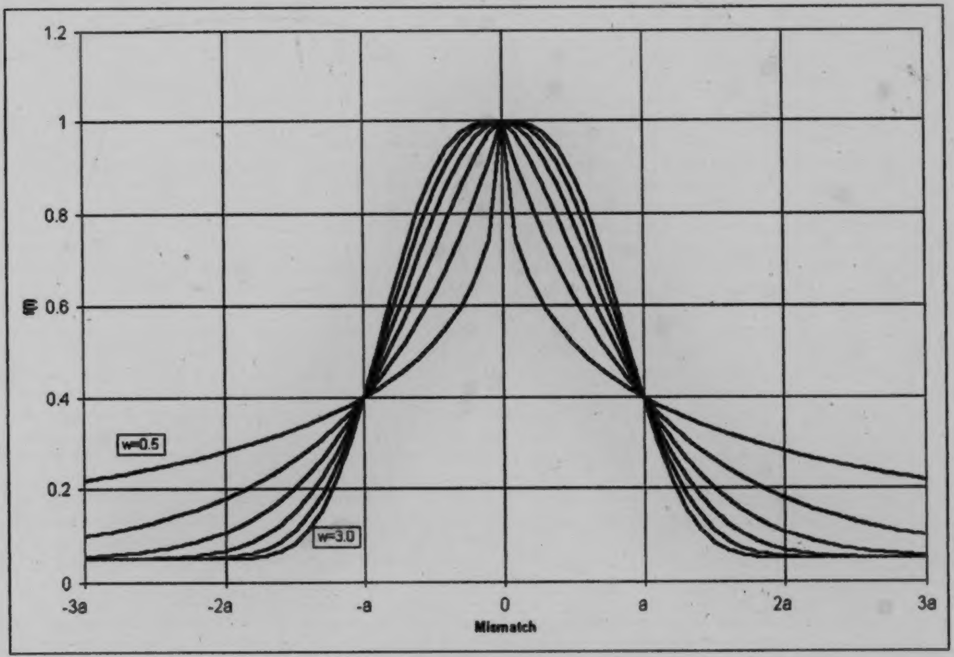


Figure 3.9: Function $f(l)$ used in the calculation of the selection function value for each candidate node to be switched. In this case, the parameter a has been set to 10 and w takes the values 0.5, 1.0, 1.5, 2.0, 2.5, and 3.0.

selection value possible (equal to one). In this case the simulation will reproduce exactly the target values.

This selection value is calculated as a product of the values taken by $f_{above}(l)$ and $f_{below}(l)$ for all length $l = 1, ..., 2 \cdot maxl$, where $maxl$ is a maximum length considered. The function has been built symmetric with respect to a mismatch of zero, because having less runs of a given length, reflected in a negative mismatch, means that the model will show a lower connectivity of nodes for that threshold than it should. This should also be avoided.

The selection function is built as a product of the exponential functions of the mismatch for different length, $f(l)$, because in this manner, if for a given length the mismatch is unacceptable (close to zero), it will significantly lower the value of $s(\mathbf{u}_i)$. If for example $s(\mathbf{u}_i)$ was built as a sum of the $f(l)$ values, a node could be selected to be switched below the threshold, that improves the matching for all lengths but one. The one length that is not matched could be critical in the performance of the realization. Considering the product of the $f(l)$ values, it is ensured that the improvement is overall better than if any other node is switched. Although the definition of what is "good" is arguable, it is considered that an improvement for all lengths of runs is better than a more significant improvement on some lengths at the cost of sacrificing the matching for other lengths.

Several shapes of the function $f(l)$ for both cases (runs above and below the threshold) are considered by means of an exponential function. These are defined by two parameters:

- Scaling parameter a affects the horizontal scale of the plot (**Figure 3.9**). The same shape is preserved as long as the other parameter, w, is kept constant; only the horizontal scale is distorted. This parameter is linked to maximum length of runs considered, $maxl$: it was found that by fixing the parameter a to be equal to $5 \cdot maxl$ gave consistent results. In this manner the algorithm also generates results that are independent on the grid definition and size of the domain simulated.

- Power parameter w changes the shape of the function $f(l)$. The higher this value, the more severely high mismatches are penalized. It also allows more fluctuations around the target values, by widening the interval of mismatches where the value of f is close to one. It was found that convergence happened for several values of this parameter. In other cases, a value for w could not be found to have a good matching of the histograms. Matching happened only for some lengths of runs, but others were always different than the target (see the examples next).

Updating Histograms of Runs

Since the selection is made by comparing the value of the selection function after switching all available nodes, one node at a time, this process is computationally very expensive. The histograms of runs must be recalculated every time. For instance, if four directions of runs are considered, and recalling that we have one histogram for runs above and one for runs below the threshold, and considering a maximum length of runs of five nodes, 40 values must be updated for every single node switched to

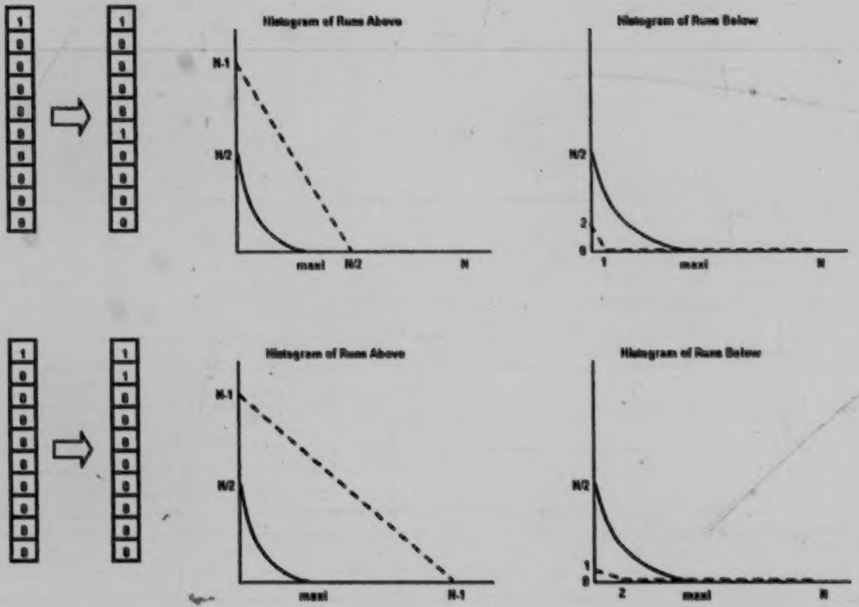


Figure 3.10: Example showing how switching a node that cuts a long run (sixth node from top to bottom on the string) gets both histograms of runs closer to the target than shortening a run (switching second node from top to bottom on the string), that is, switching one node at the end of a run, which generates a less significant change in the histograms of runs.

evaluate the selection function. Considering a grid with 10000 values, which is small for most practical applications, and a threshold corresponding to the quantile 0.90, the number of values needed quickly goes to 0.36 million. That is 0.36 million values have to be updated every time a node is switched. Since 90% of the 10000 nodes have to be switched at the highest threshold, the count goes up to the billions of values that must be recalculated only at that threshold. Multiply this by 5 or more thresholds and by the number of operations required on each calculation, and the algorithm becomes unpractical due to time considerations.

To avoid this problem, fast updating of the histograms of runs is required. The idea is then to store the data as strings, instead of individual nodes. One node will be in as many strings as directions of interest. Every time the value of a node is changed, all strings that consider that particular node are updated.

Histograms of runs are kept as partial sums of counts of runs in individual strings. Fast updating is done by analyzing the change on a node value only in the strings that contain that node, reducing considerably the amount of calculations required to update the histograms.

The counts of runs above and below the threshold for each direction are subtracted to the current histograms of runs (global) and the new counts of runs within the strings are computed and added back to the global histograms of runs.

Once the selection value is calculated, this process is reverted to get back the original global histograms of runs on each direction.

Finally, when the node to be switched has been selected, the process is repeated one last time only with that node to obtain the updated histograms of runs above and below the threshold for every direction.

3.3.2 Implementation Problems

Several problems were found during the implementation process and most were solved using rather classical solutions:

- The grid was wrapped to avoid edge effects [13, 154, 173]. The main concern was the calculation of histograms of runs. If a small grid is simulated edges will have a much larger impact than in the case of simulating a larger grid. The intention was to simulate honoring histograms of runs and to have a result independent of the grid definition.

  This is a very typical problem found in many applications. Simulated annealing shows edge effects also known as thermodynamic edge effects [41]. With neural networks, statistics on nodes close to the edges are calculated considering the nodes on the other end of the grid, that is, nodes that are adjacent when the grid is wrapped, to avoid these edge effects [13]. The same solution was applied to avoid these artifacts.

- Determining the right set of parameters w and a was done by trial-and-error. Running several examples spanning a reasonable range of parameters gave insight about the relevance of each. As mentioned before, the parameter a depends on the maximum length of runs considered. The consistency in the results seen setting $a = 5 \cdot maxl$ is due to this standardization of the penalty given to mismatches for different lengths. That is, the lengths considered when calculating the histogram of runs are treated equally for different values of *maxl*, since always the parameter a was set at the same relative value $5 \cdot maxl$. w could not be related to any parameter of the simulation and had to be found by trying different values and observing if convergence to the target histograms was reached.

- Convergence is not ensured by this method. It was found that in some cases the target histograms were not reached by the algorithm as it was. In most one dimensional examples computed, alternations around the desired proportion were necessary to ensure convergency (see for example [11]).

  Alternations can be considered as an erosion-dilation process. More nodes than necessary are pushed down (eroded), that is, their indicator values is set to one. After a given proportion of ones has been reached, larger than the target proportion, dilation starts by switching the ones to zero. Again, this is done until a proportion of zeros larger than the target is achieved. These processes of erosion and dilation are repeated getting closer and closer to the target proportion.

  Figure 3.11 shows an example where 9 alternations are used. The target proportion is 50%, that is, the threshold corresponds to the median. Each alternation goes beyond the required number to be switched, but every time it gets closer to the target proportion. The first erosion step consists on switching nodes until 90% of them are ones. This is far beyond the target of 50% of ones. Once this proportion of 90% of ones is reached, the algorithm is changed to start dilating. In practice, zeros are seen as ones and ones are seen as zeros, and the algorithm keeps eroding with the same rules than before. The first dilation corresponds to switching 80% of those ones to zero. The new proportion is now 10% of ones. Notice that the configuration of zeros and ones at this point is different than the configuration that existed when the first 10% of the nodes were eroded, that is, in the first pass of the algorithm. The new erosion process starts with 10% of ones with a spatial distribution different than the 10% of ones obtained in the first erosion. This concept is repeated to sequentially achieve 80% of ones in the second erosion, 20% of ones in the second dilation, 70% in the third erosion, 30% in the third dilation, and so forth, until the right proportion of 50% of ones is achieved after nine alternations. The key point of the process is that every time erosion and dilation has occurred, a new starting point is available.

  It is interesting to mention that some schemes of alternations made the histograms diverge from their targets, instead of converge.

- Precision problems in the calculation of the selection function value were encountered. The product of many very small numbers in **Equation 3.2** caused a problem of computer precision. When too small, these numbers are rounded down to zero, making them undistinguishable from the point of view of the selection function. This caused the algorithm to draw randomly from them, since they looked all "equal" to zero. This was solved by taking an arbitrary number of lengths, in this case $2 \cdot maxl$, where *maxl* is the maximum length on the target histograms of runs. Also, the function f has a minimum value of 0.05 to avoid values too close to 0.

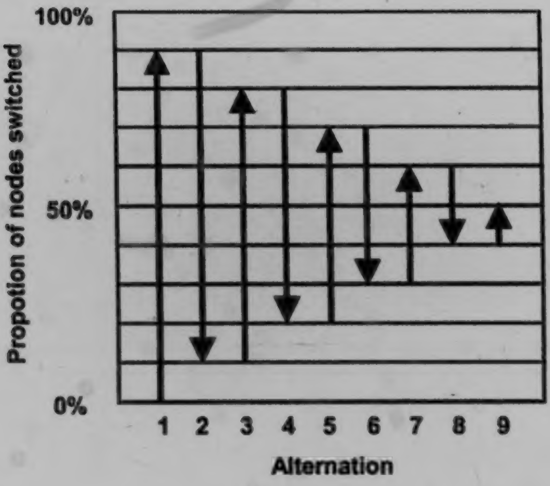


Figure 3.11: The concept of alternating to converge to the solution is showed in this schematic.

Selection functions that fixed this problem did not perform well in terms of convergence. A selection function consisting of the sum of the exponential functions of the mismatches for different lengths was tested, but the realizations never matched the target values.

3.3.3 Examples

Examples in One-Dimension

The algorithm was tested for several one dimensional cases, with one threshold. The parameters a and w are critical to ensure convergence. After a sensitivity analysis, it was found that a should be approximately the length of the simulated array, while w is dependent on the complexity of the problem. All of the examples presented worked with $w = 4.6$.

Random Case The first case consists of simulating the runs found on random sequences, when coded as above or below a threshold.

A reference sequence of random numbers was generated with the random number generator `acorni`. The numbers generated were then coded as 1 if below or equal 0.5, and 0 if above 0.5.

The reproduction of the histograms of runs above and below the threshold, the reference string of indicator values, and five realizations are shown in **Figure 3.12**. Some fluctuations around the target values are observed. The algorithm tends not to reproduce the runs of length one, that is, isolated white or black nodes. The realizations look more continuous than the reference, which has no spatial continuity by construction. It can be seen in the histogram of runs below the median, that

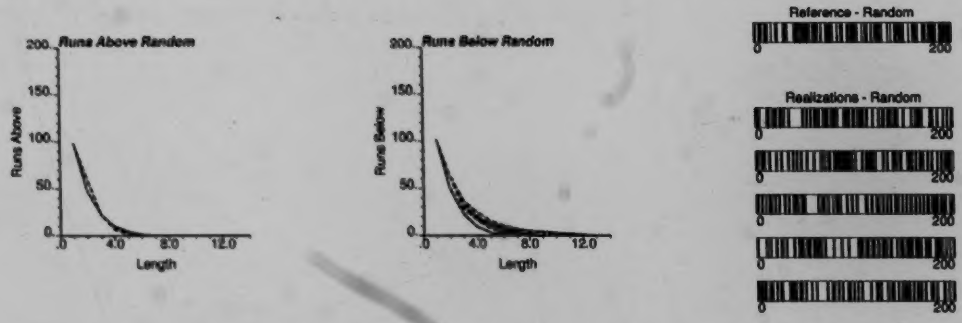


Figure 3.12: Reproduction of runs above and below the median for a random sequence.

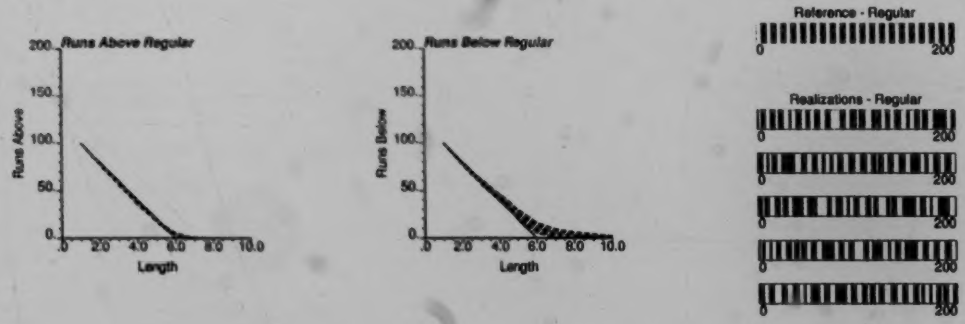


Figure 3.13: Reproduction of runs above and below the median for a regular sequence.

some long runs (in white) are generated. Also, notice that the histogram of runs above the threshold is better reproduced than the histogram of runs below it.

Regular Case A regular array with sets of five nodes above and five below the threshold was used as a reference to test the algorithm. The reproduction of the histograms of runs, reference string of data, and five realizations are shown in **Figure 3.13**. As with the random case, the histogram of runs above the threshold seems to be better reproduced. This could be caused by the alternation sequence that tends to give more importance to the histogram above, since the differences are larger (see **Figure 3.8**). Some long runs below the threshold are generated and some of the runs above it are broken into shorter ones.

Multi-Gaussian Case A one dimensional array was simulated using the algorithm `sgsim` of GSLIB [39]. The realization was then truncated at the median to generate a binary array. The histograms of runs extracted from it was simulated using the algorithm proposed. The resulting histograms of runs, reference and simulated strings are presented in **Figure 3.14**. Notice the apparent good reproduction obtained from the histograms. Again, when looking at the realizations all runs of

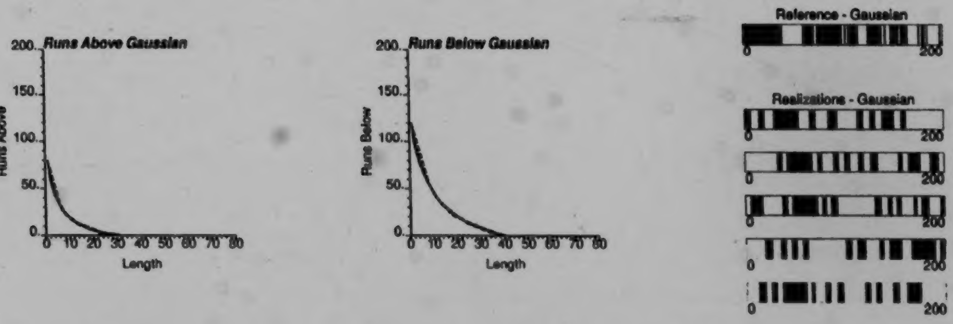


Figure 3.14: Reproduction of runs above and below the median for a binary array obtained by truncating a multi-Gaussian sequence.

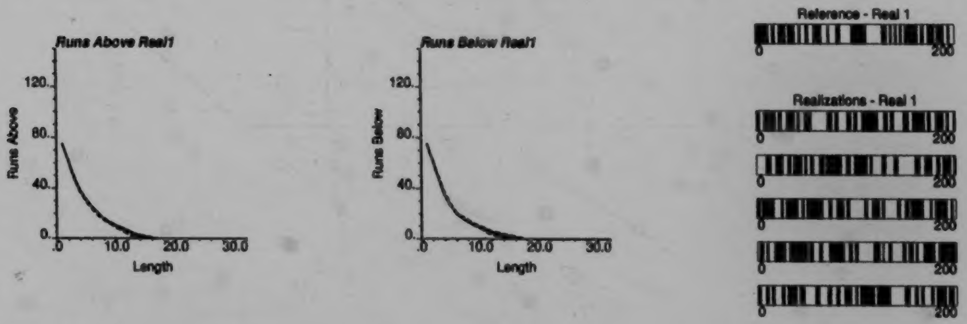


Figure 3.15: Reproduction of runs above and below the median for a binary array obtained from a realistic exhaustive data set.

length 1 seem to be lost. Visually, these realizations appear quite different than the reference string. All the noise of the reference image is lost in the different renditions obtained through this algorithm.

Real Data 1 A string from an exhaustive data set was used to obtain the multiple-point runs statistics. The resulting simulated sequences showed very good reproduction of the reference statistics as seen on **Figure 3.15**. In this case very few short runs existed in the reference. The algorithm generated realizations that look similar to the reference string of indicator values, although short runs are again lost.

Real Data 2 A second example from real data was computed. The histogram of runs was again very well reproduced and the realizations look similar to the reference (**Figure 3.16**) with short runs missing.

Extension to 2D and 3D

Extending the algorithm to two and three dimensions is straightforward. Several directions of runs can be considered at the same time. The selection function will

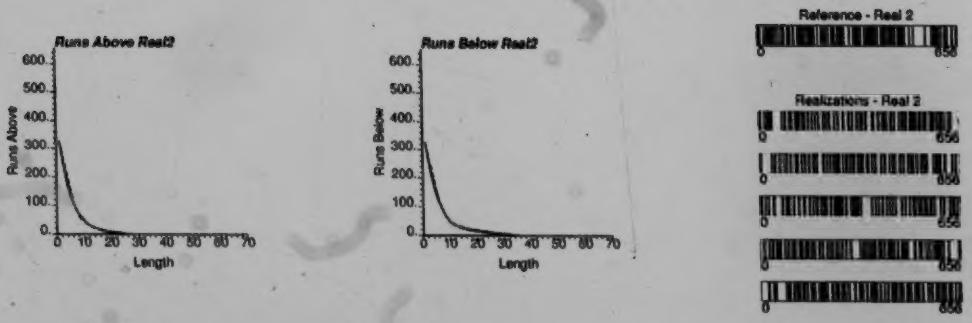


Figure 3.16: Reproduction of runs above and below the median for a binary array obtained from a second realistic exhaustive data set.

be calculated as the value of the function $f(l)$ for several directions. For example, working with two directions, the selection function corresponds to the product of four values of the $f(l)$ function: two corresponding to the mismatches for runs above and below the threshold for the first direction of runs, and the other two corresponding to the second direction.

Several examples in two and three dimensions were computed.

Pure Nugget Effect A variable that has no spatial correlation will generate a variogram with a pure nugget effect, that is, all variability is due to pure randomness. In this case, the frequencies of runs above and below any threshold are easily computed and are a function of the proportion above and below the threshold only (see **Section 3.2**). A 40 by 40 nodes grid was used. **Figure 3.17** shows indicator maps for a reference realization . **Figures 3.18** and **3.19** show the indicator maps for two realizations generated with the proposed algorithm. The first one aims to reproduce runs up to a maximum length of 3, while the second one uses 8 as a maximum length. Runs are considered above and below every one of the nine deciles of a standard Gaussian distribution in four directions with azimuths of 0, 45, 90, and 135 degrees.

Undesired patterns are clearly seen in both realizations, particularly at low thresholds (the first ones being simulated). They are due to the selection function used, which does not give an equal probability to every node in the grid to be set above the current threshold. **Figure 3.20** shows the indicator variograms for the model generated using runs up to a maximum length of 3; **Figure 3.21** shows the result when a maximum length of 8 is used. The dashed lines correspond to the variograms in the horizontal and vertical directions for the reference image, the solid lines are the corresponding variograms for the simulated models. In the first case, the indicator variograms do not reflect the artifact that can clearly be seen on the indicator maps. In the second case, a higher correlation (lower variogram value) is seen at some lag distances, in particular 5 and 10 units. These can be attributed to the selection function. It can be seen that the spatial correlation disappears as the thresholds move upwards. The final result is shown on **Figure 3.22**.

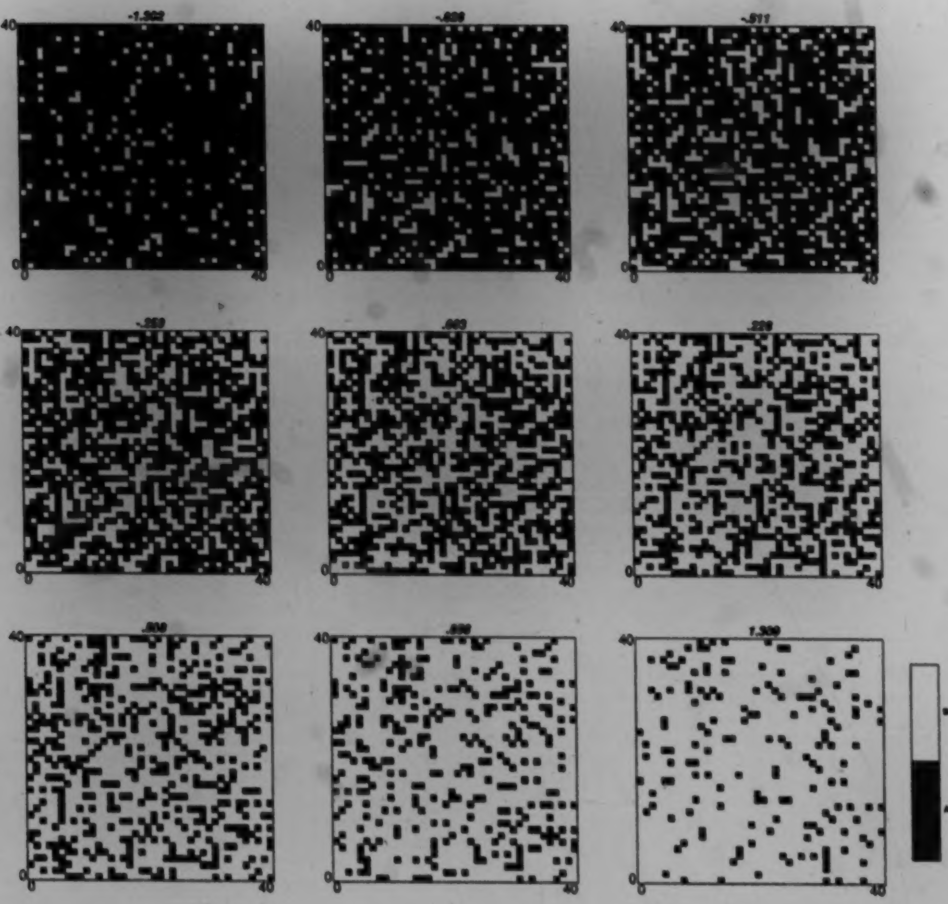


Figure 3.17: Indicator maps for a spatially uncorrelated variable.

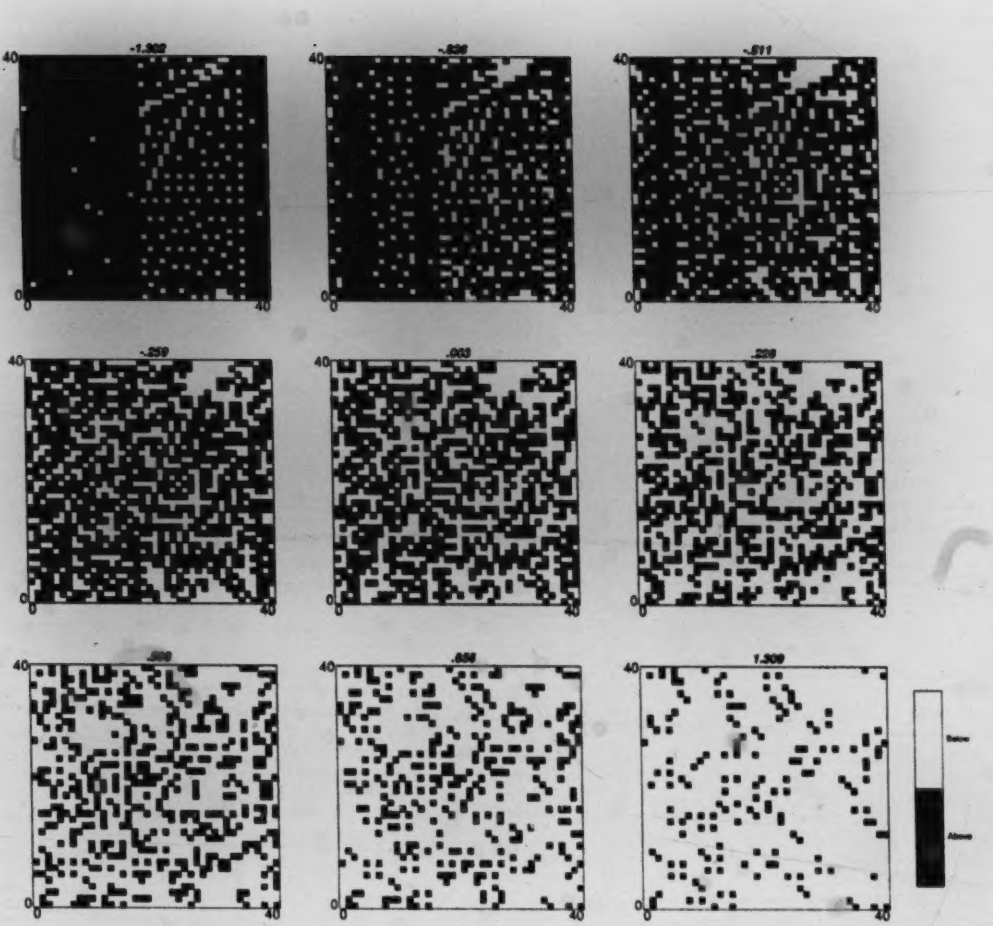


Figure 3.18: Random case: indicator maps for a simulated model using a maximum length of runs of 3.

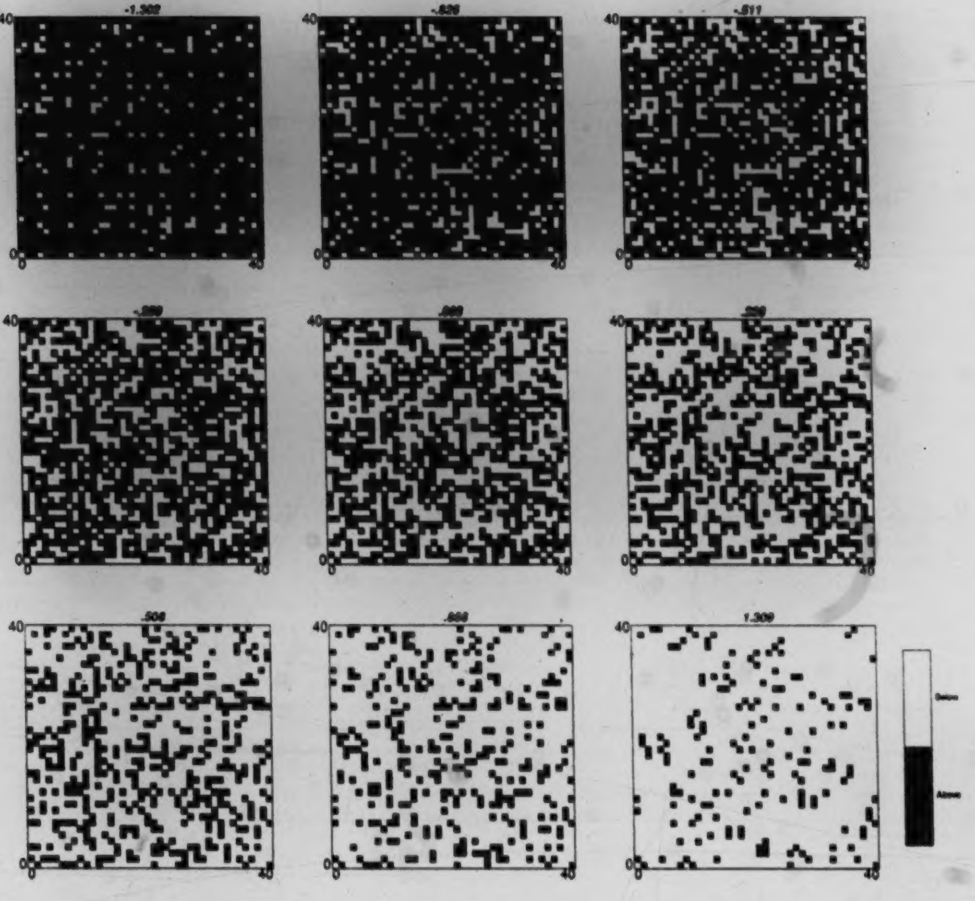


Figure 3.19: Random case: indicator maps for a simulated model using a maximum length of runs of 8.

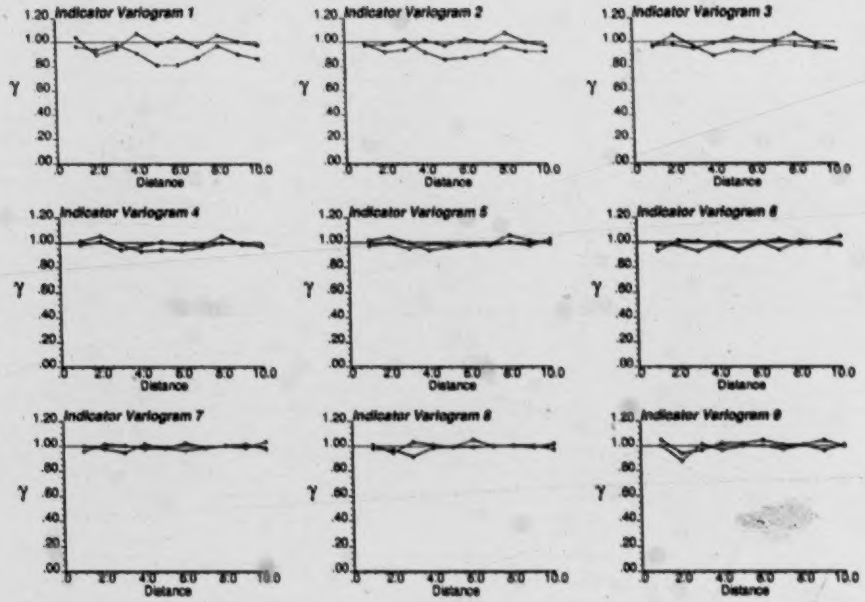


Figure 3.20: Random case: indicator variograms for a simulated model using a maximum length of runs of 3.

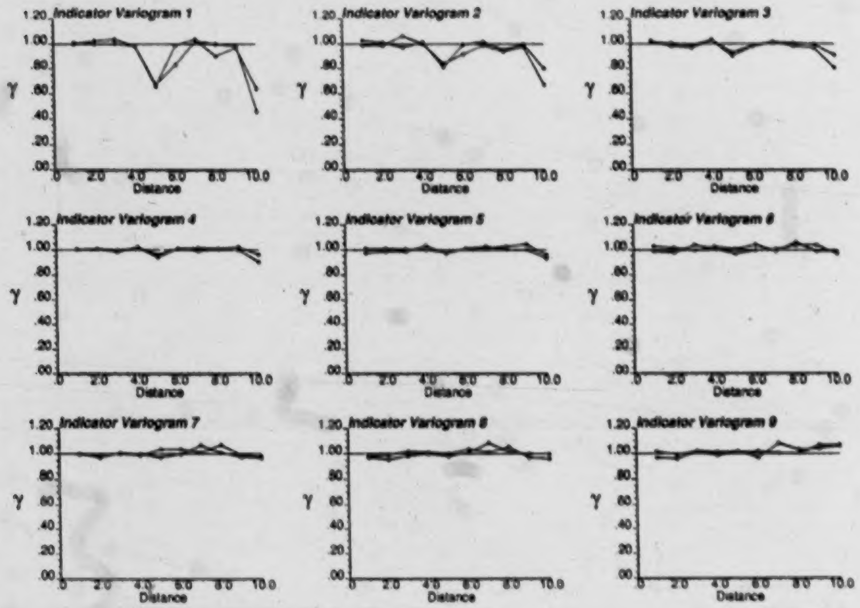


Figure 3.21: Random case: indicator variograms for a simulated model using a maximum length of runs of 8.

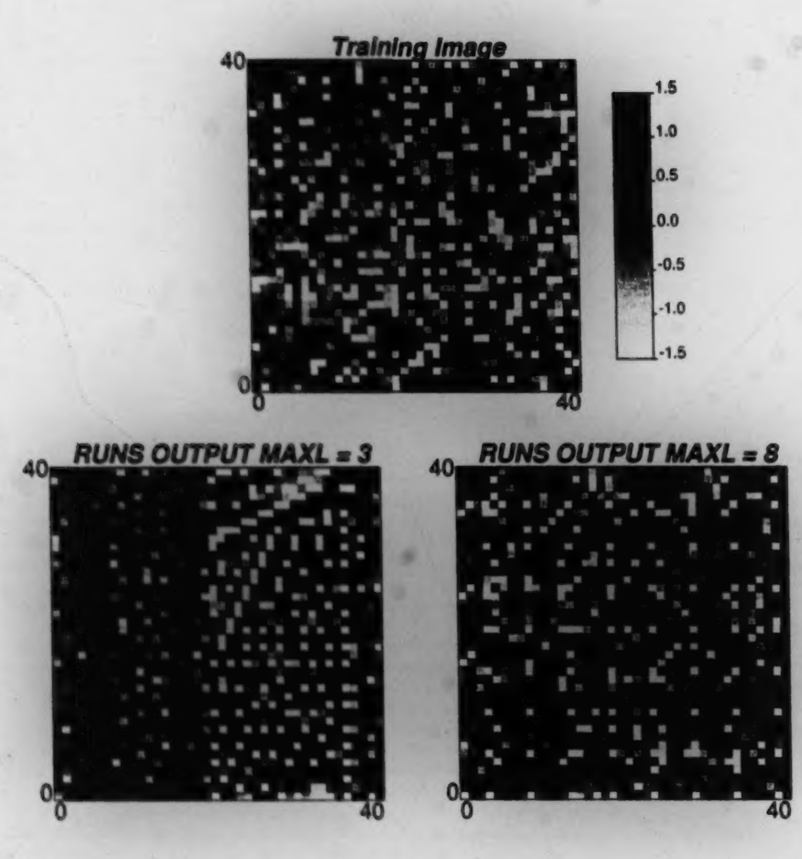


Figure 3.22: Random case: maps of the training image and the simulated models with maximum length of 3 and 8.

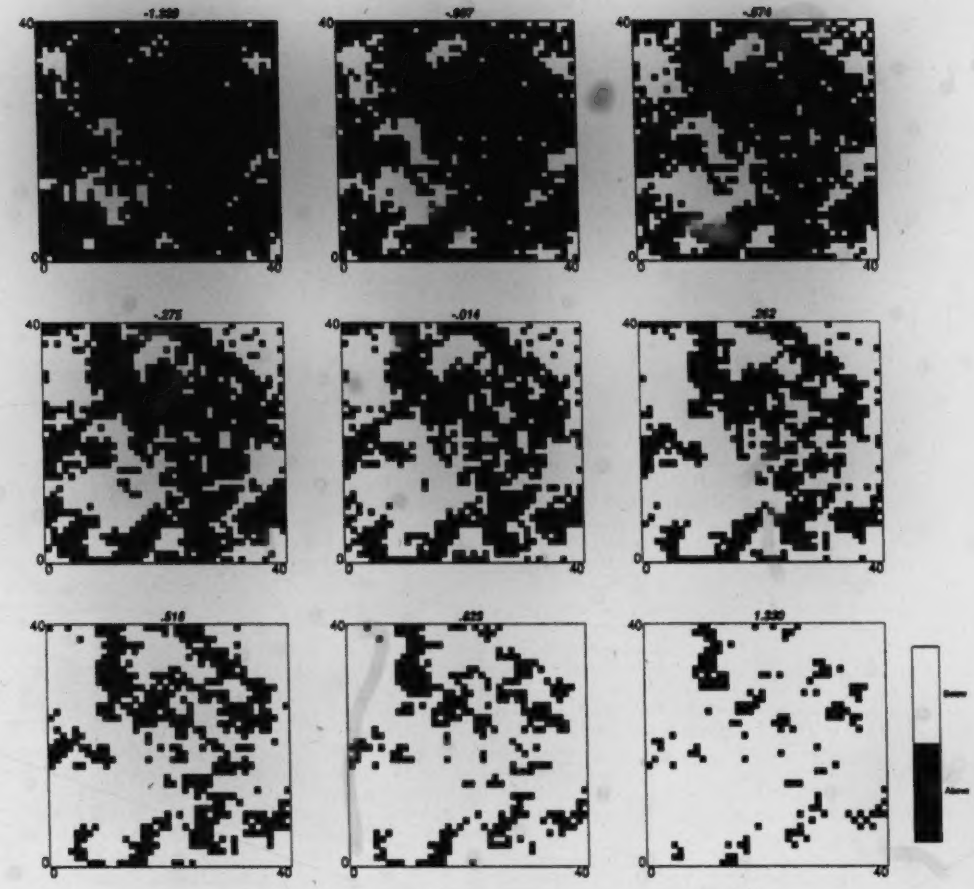


Figure 3.23: Indicator maps for a multivariate Gaussian correlated variable.

Multi-Gaussian Case A second example was built using a multi-Gaussian random function. An unconditional realization was generated (**`sgsim`**) as a reference (or training) image. Frequencies of runs and threshold values were extracted from it. The variogram model used was an isotropic spherical model with a range of 5 units and a 10% of nugget effect. As in the previous case, indicator maps were constructed for the nine deciles of the distribution and for each one of two realizations built accounting for the frequencies of runs considering maximum lengths of runs of 4 and 8 units. These are presented in **Figures 3.23**, **3.24**, and **3.25**. Again, a regular pattern appears at the lowest threshold with a spacing of 5 units, which is clearly reflected in the indicator variograms computed for each case (**Figures 3.26** and **3.27**). The simulated models present a higher nugget effect than the target indicator variograms, however the shape of the structures appears quite similar to the ones seen in the reference image.

Finally, the reference image along with the two generated accounting for runs up to different lengths are presented on **Figure 3.28**. The noise added at the early stages of the simulation can be seen on the final models.

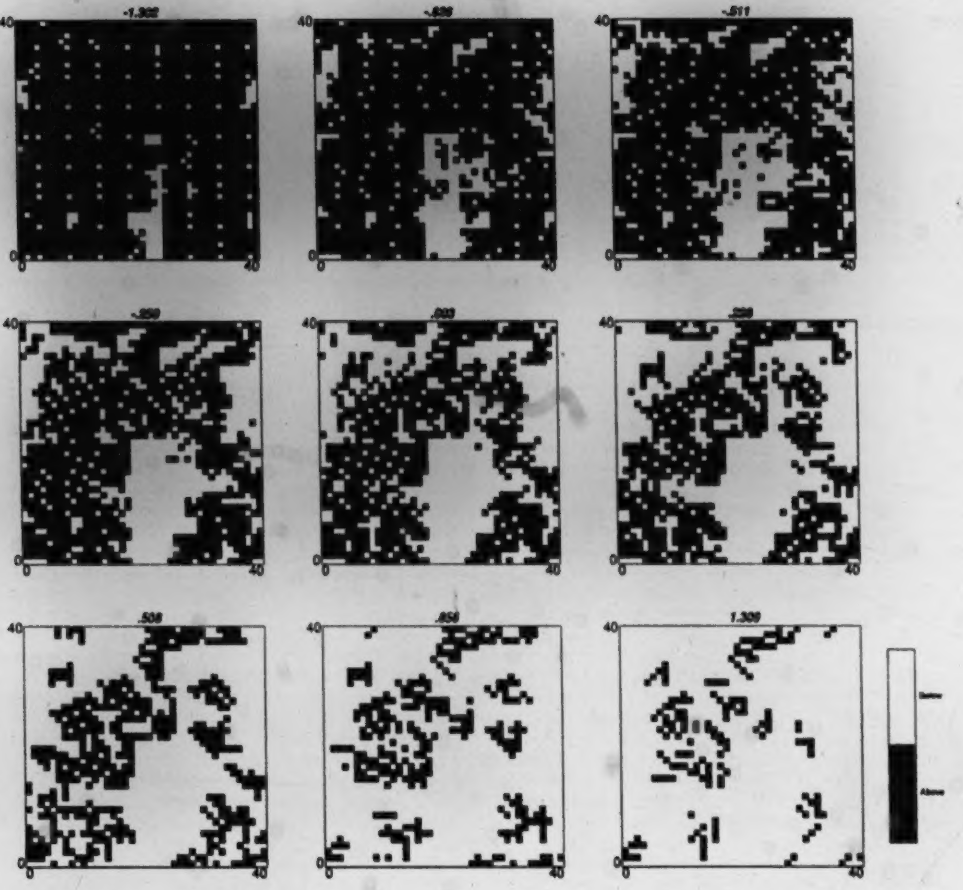

-1.302
-.636
-.511
-.250
.003
.236
.508
.856
1.309
0
40
Below
Above

Figure 3.24: Multi-Gaussian case: indicator maps for a simulated model using a maximum length of runs of 4.

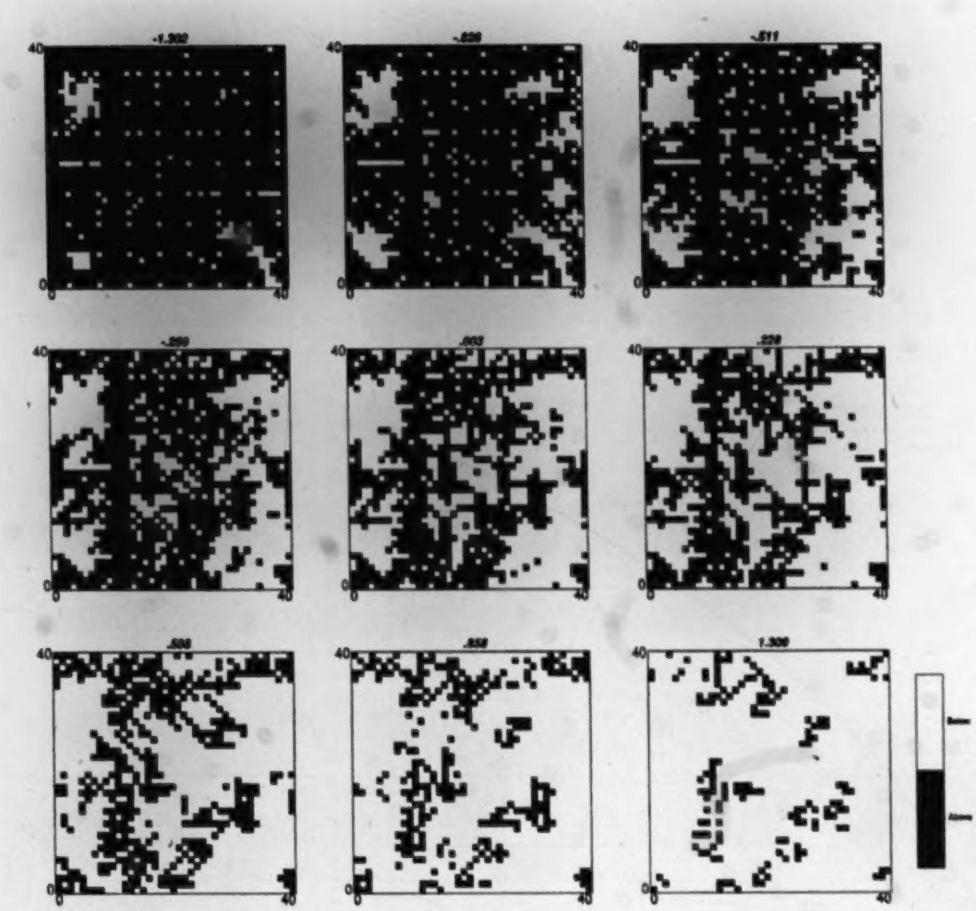

-1.302
-.828
-.511
-.290
.003
.228
.588
.858
1.300
40
0
0
40

Figure 3.25: Multi-Gaussian case: indicator maps for a simulated model using a maximum length of runs of 8.

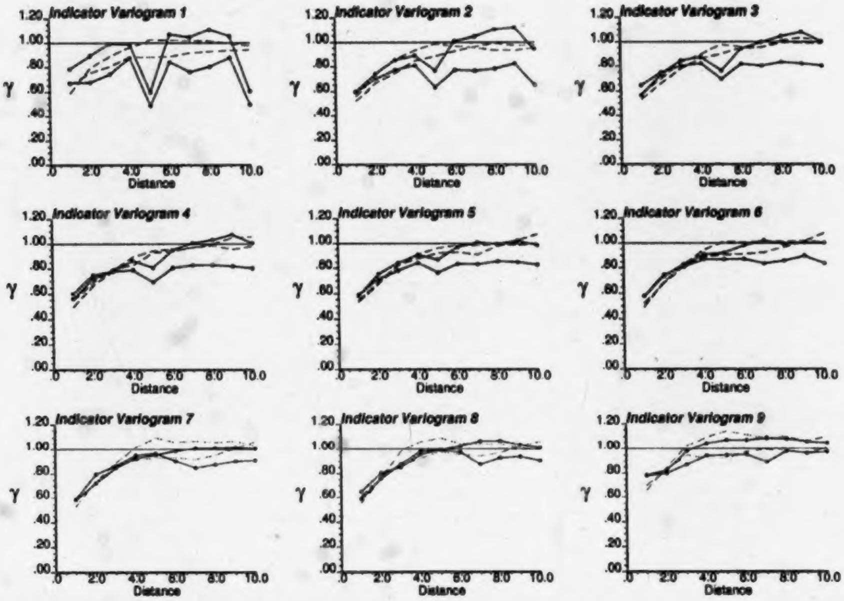


Figure 3.26: Multi-Gaussian case: indicator variograms for a simulated model using a maximum length of runs of 4.

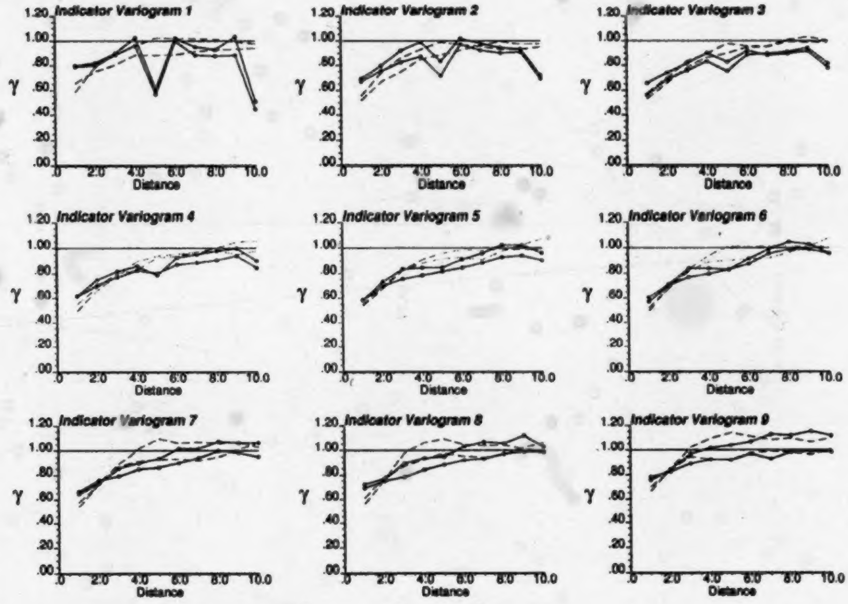


Figure 3.27: Multi-Gaussian case: indicator variograms for a simulated model using a maximum length of runs of 8.

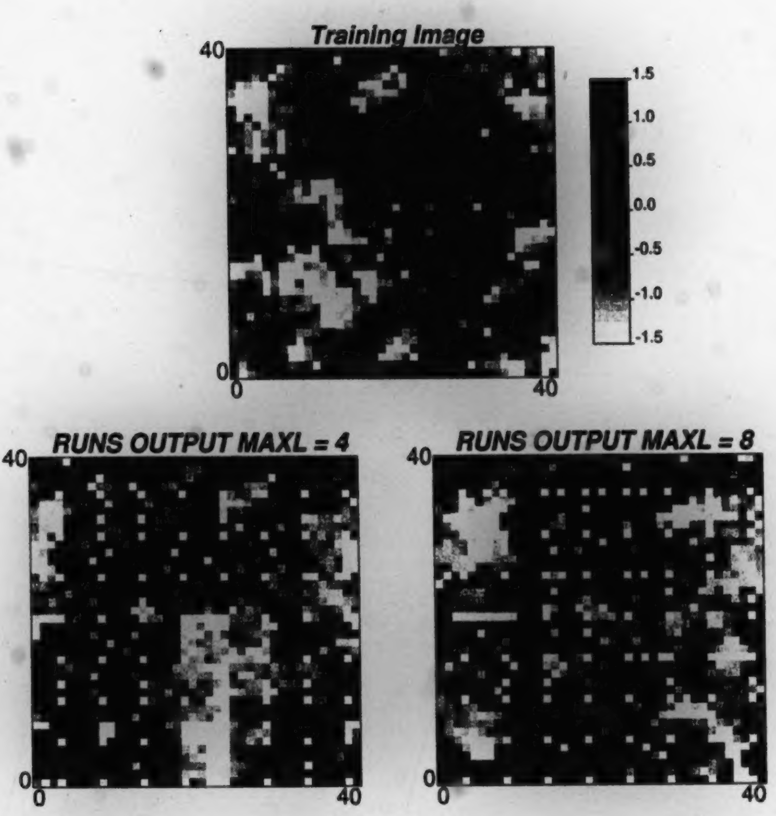

Training Image
40
0
0
40
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
RUNS OUTPUT MAXL = 4
40
0
0
40
RUNS OUTPUT MAXL = 8
40
0
0
40

Figure 3.28: Multi-Gaussian case: maps of the training image and the simulated models with maximum length of 4 and 8.

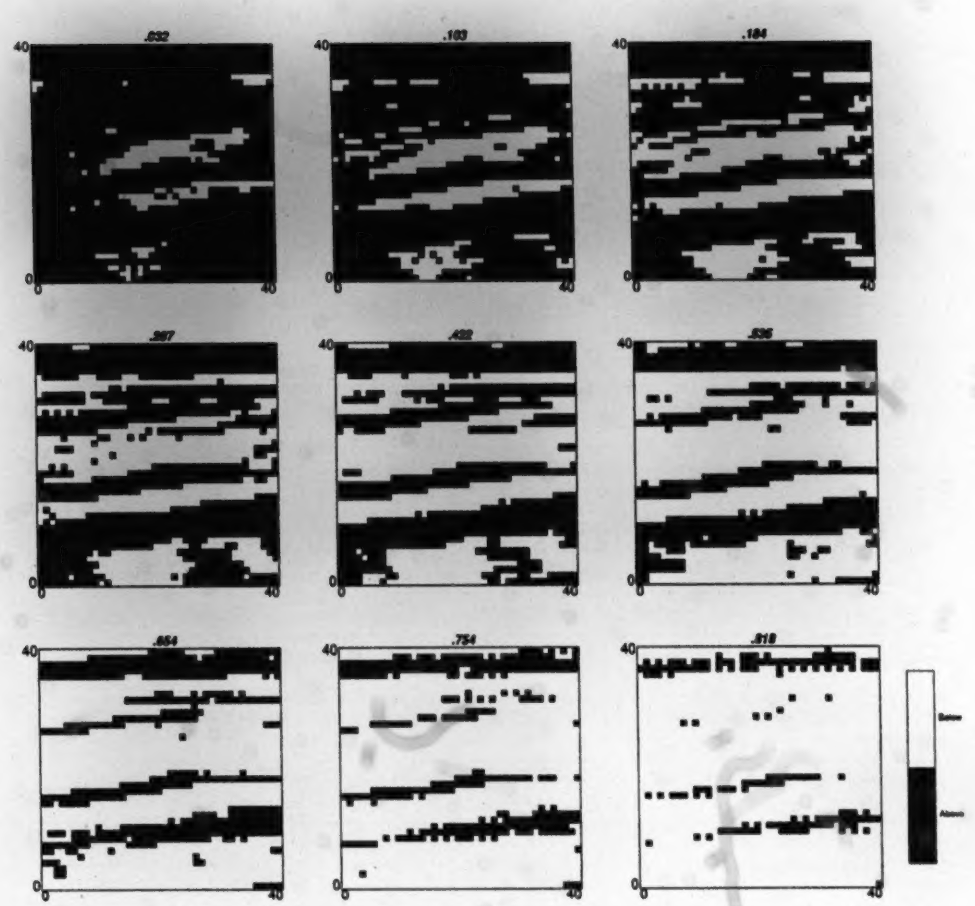


Figure 3.29: Indicator maps for real data.

Case using Real Data An exhaustive training image obtained from a section of a rock collected at a sedimentary deposit is used in this example. The exhaustive image is used to extract the multiple-point statistics and histogram. The frequencies of runs above and below the nine deciles of the distribution were obtained. The indicator maps for the reference image and a simulated model considering runs up to a length of 10 units in the horizontal direction and up to 5 in the vertical and two diagonal directions, are presented in **Figures 3.29** and **3.30**. The indicator variograms obtained are presented in **Figure 3.31**. The matching is not good.

Finally, the maps of the reference image and simulated one are presented in **Figure 3.32**.

Computation Time

Some of the runs were timed to find out the impact of increasing the model size and considering longest runs. Around two minutes are necessary to simulate a 40 by 40 nodes model using four directions, nine thresholds, and runs of up to 8 nodes in all directions.

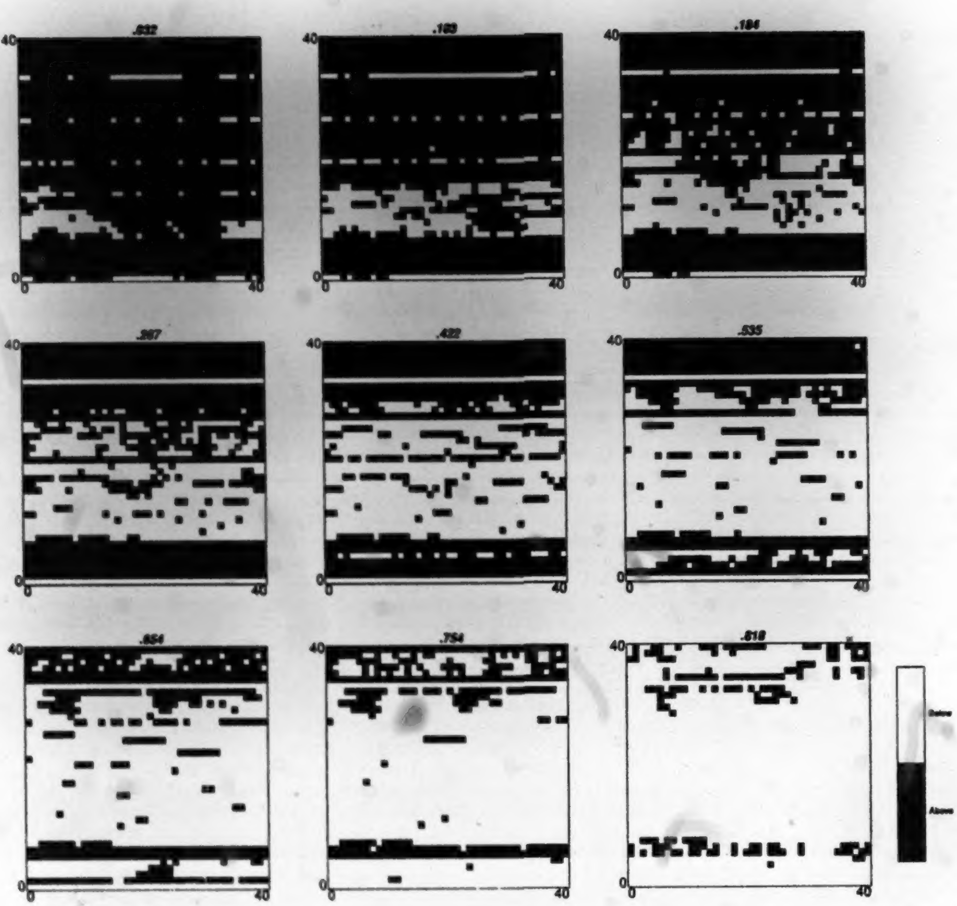


Figure 3.30: Case with real data: indicator maps for a simulated model.

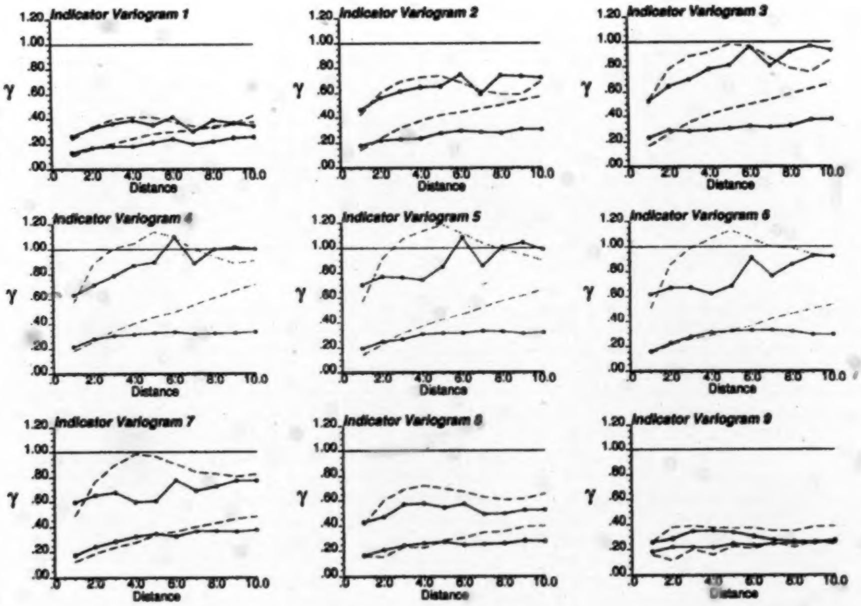


Figure 3.31: Case with real data: indicator variograms for a simulated model.

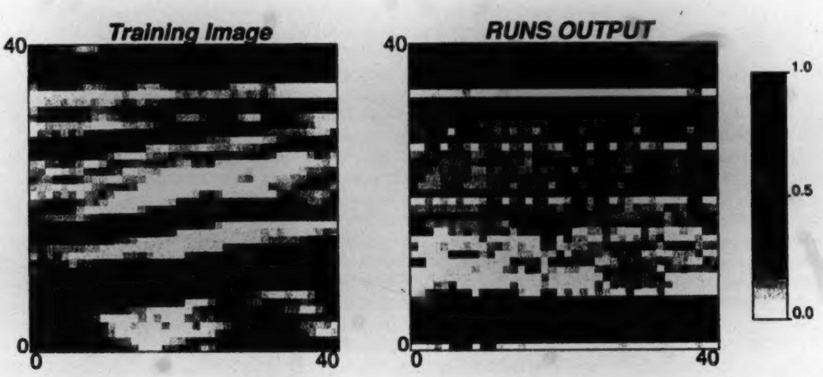

Training Image
RUNS OUTPUT
40
0
0
40
1.0
0.5
0.0

Figure 3.32: Case with real data: maps of the training image and the simulated model.

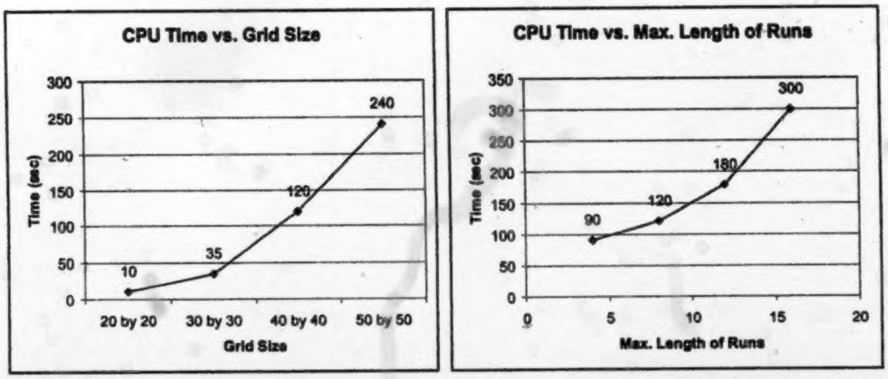


Figure 3.33: CPU time required to run a model with nine thresholds and considering four directions for the multiple-point runs. Left: Models of 20 by 20 nodes, 30 by 30 nodes, 40 by 40 nodes, and 50 by 50 nodes are computed considering a maximum length of runs of 8. Right: Runs up to a maximum length of 4, 8, 12, and 16 are considered on a 40 by 40 nodes model.

Sensitivities with respect to the maximum length of runs considered and the grid size can be seen in **Figure 3.33**.

For large grids computation time would make this algorithm unpractical, even if the artifacts were corrected.

3.4 Comments on the Direct Simulation of Runs

The implementation has a few problems that have been partially addressed. Firstly, there is a problem with the selection criterion, since in the case of pure nugget effect, all nodes should always have the same probability of being switched to a lower threshold. This is not happening. Unwanted structure exists in all the models that does not belong to the phenomenon being simulated.

Convergence is another major issue. The use of heuristic scaling parameters to make the simulation converge is not appealing. Although rules for these parameters could be found by sensitivity analysis, they do not have a clear meaning, making them very difficult to interpret. Problems found in one-dimension were not apparent in two dimensions. The alternating approach was not needed in the 2-D examples.

This is a type of optimization problem. The more restrictions applied to the problem, by defining more directions of interest and longer runs, the more difficult it gets to converge. Depending on the consistency of the data used to infer these runs, a solution may not even exist. This can be better explained by considering a combination of runs in one direction that restricts physically the existence of runs in another direction.

Given all the previous considerations, this approach was discarded, although some insight is given and it is considered valuable.

Runs do not fully characterize the multivariate spatial distribution of the variable, that means that although in some cases the histograms of runs were closely reproduced, the simulated images did not look like the reference, since further multiple-point features were not captured by this statistic.

The idea of erosion requires the algorithm to take into account the probabilities of being below the threshold. This was not achieved properly by the proposed algorithm. In the case of trying to reproduce a variable randomly distributed in space, that is, without spatial correlation, the probability of eroding a node should have been the same for all nodes in the field, however the algorithm privileged some nodes, due to the selection function considered. This is what caused the generation of undesired regular patterns and invalidated the method.

An alternative approach is proposed next, by considering updating of the probabilities obtained through indicator kriging. This algorithm improves the performance of numerical models and is simpler to apply.

Chapter 4

Updating the Indicator Kriging Probability with Multiple-Point Statistics

This chapter discusses the implementation of updating techniques to merge indicator kriging probabilities and multiple-point statistics. These techniques are presented as an approach to simulate with multiple-point statistics and their implementation improves results given by conventional sequential indicator simulation.

Section 4.1 reviews several ways to integrate information from multiple sources. A new approach is proposed. **Section 4.2** explains the difference between a kriging-like approach to integrate data and the framework of updating probabilities.

Section 4.3 presents different approaches to update the probabilities obtained by indicator kriging (IK) and multiple-point (MP) statistics. These assumptions are presented in generality.

Practical implementation details are presented in **Section 4.4**. The IK probabilities are updated using MP statistics obtained for some particular configurations of points.

Examples are given in **Section 4.5**. The improvements on model performance are assessed in **Section 4.6** by considering statistical measures such as the mean squared error of the multiple-point probabilities.

Section 4.7 shows an analysis of the non-convexity of the different estimators. These values provide some insight about the performance of the methods.

Finally, a brief discussion about the results and the methods is presented in **Section 4.8.**

4.1 Introduction

Integration of MP statistics into geostatistical models is difficult because the inference of these statistics is often unreliable and their use in a kriging-like framework requires the positive definite calculation or modelling of covariances between MP events and single-point events that calls for the knowledge of the multivariate distribution of the variable.

One way to overcome these problems is to use a training image deemed representative of the phenomenon being studied [7, 15, 77, 157, 158]. This raises different problems: the representativeness of this training image, the scale of the training image, the grid definition of the simulated model, and the univariate distribution of the training image. A concern is the amount of information deemed general to the phenomenon and the amount considered particular to the training image. The goal is not to reproduce the training image, but to simulate a model that shares some of the multivariate characteristics of the true value, represented by this training image.

The paradigm of the extended normal equations and the consequent concept of the single normal equation solve most of these problems [76, 157]. A conditional distribution function is calculated that considers the MP configuration in a neighborhood. The probability of that particular location to be above a threshold is given by the frequency of occurrence of that MP configuration in the training image. Therefore, there is no kriging system to solve. The multivariate distribution is being approximated by the frequencies extracted from the training image.

The question of the representativeness of the training image remains. The only apparent way out of this problem would be to use MP statistics extracted from the actual data [132]. Borrowing the MP information from a training image is equivalent to using the variogram from a different area, deposit, or reservoir to build a numerical model that reproduces the spatial continuity of the phenomenon. Expert judgement is used.

In most geostatistical techniques the multivariate distribution is commonly multivariate Gaussian. Consequently, high order and non-linear connectivity is not reproduced in the numerical model. The response after a transfer function is not reliably reproduced. [65]. Transfer functions can be sensitive to high order correlation, that is, continuity of high and low values in the model. In mining, the mine design, mine plan, and grade control could change as the high order correlation is better reproduced. The design of stopes and open pits may also change as a consequence of the multiple-point characteristics of the numerical model used. Better reproduction of multiple-point statistics at point support may allow block averaged values to follow more closely the true block distribution, improving grade control.

If the data present a multivariate distribution that departs from multi-Gaussian, multiple-point statistics have to be explicitly imposed in the simulation algorithm to control these high order spatial relationships. In order to extract statistics from the data and avoid modelling of the high-order covariances, some updating techniques are considered to improve the reproduction of MP features [90]. The multivariate distribution is pushed closer to the one of the data used to extract these MP statistics, although not explicitly honored. The updating is done in the indicator framework. The idea is to update the conditional probabilities calculated by indicator kriging with the ones based on multiple-point configurations. This is similar to what is done in collocated cokriging. It can be seen as a Bayesian updating of the probability provided by indicator kriging [48, 49, 50].

A short discussion on the inference of two-point and multiple-point statistics is presented. Then, the idea of directly drawing from the multiple-point probability, disregarding the one obtained by indicator kriging, is presented, and finally, several updating techniques to combine both sources of information are discussed.

4.2 Statistical Inference of Two-Point and Multiple-Point Statistics

The expansion of current methods toward the use of higher order statistics has been impeded by the problem of inferring these statistics. Many practitioners find inference of second-order moments (covariances or variograms) quite challenging in presence of relatively sparse data. The inference and modeling of more complex moments is daunting. The problem is exacerbated by the further requirement that all statistics be jointly positive definite.

Updating techniques do not require an explicit model. Consider, for example, collocated cokriging. The prior is given by the local distribution obtained by kriging the primary data only. The likelihood distribution corresponds to the distribution generated from the secondary data. A posterior distribution is generated that corresponds to the updated one by Bayes' law, exactly equivalent to collocated cokriging. This model is consistent. The Markov hypothesis and the use of the linear correlation coefficient provide a consistent model of coregionalization, without the need of difficult modelling under the linear constraints.

4.3 Integrating Multiple-Point Statistics

The probability of a variable Z to exceed a threshold z_k at location $\mathbf{u}$ is of interest, which is called event $\mathbf{A}$. A number of events R that inform this location is available to calculate the conditional probability of $\mathbf{A}$ at $\mathbf{u}$. These events are as $\mathbf{B}_1, ..., \mathbf{B}_R$. They may correspond to any arrangement of any number of data at any support. They can be disjoint or have elements in common. They can be considered as sets of elements, such as the samples used in kriging to estimate the value at an unsampled location, or they can be considered as a joint event, such as a multiple-point event, that is, a configuration of samples with fixed values.

Consider the case where information from several different sources is used to estimate the conditional probability of event $\mathbf{A}$. Bayes' law gives a formalism to calculate this conditional probability. These different sources of information can be integrated to estimate the posterior conditional probability. The integration of multiple sources of information, however, calls for a model of redundancy between these sources.

Bayes' law gives the general expression for the conditional probability of the event $\mathbf{A}$:

$$P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_R) = \frac{P(\mathbf{A}, \mathbf{B}_1, ..., \mathbf{B}_R)}{P(\mathbf{B}_1, ..., \mathbf{B}_R)} \tag{4.1}$$

However, this expression requires the knowledge of the joint distribution of the events $\mathbf{B}_1, ..., \mathbf{B}_R$ with event $\mathbf{A}$, that is, $P(\mathbf{A}, \mathbf{B}_1, ..., \mathbf{B}_R)$. These multivariate distributions are difficult to infer, thus their use is avoided in practice.

The conditional probability $P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_R)$ is estimated by making some assumption about the relationship between the different sources of information.

Recursive application of Bayes' law permits **Equation 4.1** to be rewritten as:

$$P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_R) = \frac{P(\mathbf{B}_R|\mathbf{A}, \mathbf{B}_1, ..., \mathbf{B}_{R-1}) \cdot P(\mathbf{B}_{R-1}|\mathbf{A}, \mathbf{B}_1, ..., \mathbf{B}_{R-2}) \cdots P(\mathbf{B}_1|\mathbf{A}) \cdot P(\mathbf{A})}{P(\mathbf{B}_1, ..., \mathbf{B}_R)}$$

This will simplify calculation later when assumptions are made about the redundancy of the sources of information.

4.3.1 Assumption of Independence Between Multiple Events

The assumption of independence between $\mathbf{B}_1, \mathbf{B}_2, ..., \mathbf{B}_R$ permits the calculation of the conditional probability of $\mathbf{A}$ as the product of the independent probabilities:

$$P(\mathbf{B}_1, ..., \mathbf{B}_R) = P(\mathbf{B}_1) \cdot P(\mathbf{B}_2) \cdots P(\mathbf{B}_R)$$

The expression for the conditional probability of $\mathbf{A}$ given $\mathbf{B}_1, ..., \mathbf{B}_R$ is simplified to:

$$P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_R) = \frac{P(\mathbf{A}|\mathbf{B}_1)}{P(\mathbf{A})} \cdot \frac{P(\mathbf{A}|\mathbf{B}_2)}{P(\mathbf{A})} \cdots \frac{P(\mathbf{A}|\mathbf{B}_R)}{P(\mathbf{A})} \cdot P(\mathbf{A})$$

This simplification comes from assuming conditional independence of the likelihoods, with respect to the conditioning event $\mathbf{A}$:

$$P(\mathbf{B}_i|\mathbf{A}, \mathbf{B}_1, ..., \mathbf{B}_{i-1}) = P(\mathbf{B}_i|\mathbf{A}) \qquad \forall i = 2, ..., R$$

and from Bayes' law:

$$P(\mathbf{B}_i|\mathbf{A}) = \frac{P(\mathbf{A}|\mathbf{B}_i)}{P(\mathbf{A})} \cdot P(\mathbf{B}_i)$$

Independence between the events $\mathbf{B}_1, ..., \mathbf{B}_R$ appears as a very strong assumption in our context, since the variable is certainly spatially correlated. It seems unrealistic to have the events $\mathbf{B}_1, ..., \mathbf{B}_R$ correlated to $\mathbf{A}$, and yet have them uncorrelated with respect to each other.

4.3.2 Permanence of Ratios Assumption

The assumption of permanence of ratios is another way around the problem of knowing the joint probability of $\mathbf{B}_1, ..., \mathbf{B}_R$ [90]. This assumption basically states that the incremental information provided by one event $\mathbf{B}_i$ before and after knowing the others is constant. Although not as strong as the assumption of independence between all the events $\mathbf{B}_1, ..., \mathbf{B}_R$, this assumption also calls for a model of dependence that could be proven wrong. However, practice has shown that it performs better than the previous independence assumption.

The permanence of ratios assumption can be written by considering first a pair of events $\mathbf{B}_1$ and $\mathbf{B}_2$:

$$\frac{\frac{1-P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2)}{P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2)}}{\frac{1-P(\mathbf{A}|\mathbf{B}_1)}{P(\mathbf{A}|\mathbf{B}_1)}} = \frac{\frac{1-P(\mathbf{A}|\mathbf{B}_2)}{P(\mathbf{A}|\mathbf{B}_2)}}{\frac{1-P(\mathbf{A})}{P(\mathbf{A})}}$$

From this expression, the conditional probability $P(\mathbf{A}|\mathbf{B}_1, \mathbf{B}_2)$ can be calculated:

$$P(\mathbf{A}|\mathbf{B}_1, \mathbf{B}_2) = \frac{\frac{1-P(\mathbf{A})}{P(\mathbf{A})}}{\frac{1-P(\mathbf{A})}{P(\mathbf{A})} + \frac{1-P(\mathbf{A}|\mathbf{B}_1)}{P(\mathbf{A}|\mathbf{B}_1)} \cdot \frac{1-P(\mathbf{A}|\mathbf{B}_2)}{P(\mathbf{A}|\mathbf{B}_2)}}$$

Now considering a third event $\mathbf{B}_3$, the conditional probability $P(\mathbf{A}|\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3)$ can be retrieved by using the previous expression for $P(\mathbf{A}|\mathbf{B}_1, \mathbf{B}_2)$. First, the permanence of ratios is established:

$$\frac{\frac{1-P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2,\mathbf{B}_3)}{P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2,\mathbf{B}_3)}}{\frac{1-P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2)}{P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2)}} = \frac{\frac{1-P(\mathbf{A}|\mathbf{B}_3)}{P(\mathbf{A}|\mathbf{B}_3)}}{\frac{1-P(\mathbf{A})}{P(\mathbf{A})}}$$

Now, rearranging, the expression of interest can be found:

$$P(\mathbf{A}|\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3) = \frac{\frac{1-P(\mathbf{A})}{P(\mathbf{A})}}{\frac{1-P(\mathbf{A})}{P(\mathbf{A})} + \frac{1-P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2)}{P(\mathbf{A}|\mathbf{B}_1,\mathbf{B}_2)} \cdot \frac{1-P(\mathbf{A}|\mathbf{B}_3)}{P(\mathbf{A}|\mathbf{B}_3)}}$$

Similarly, the permanence of ratios assumption can be generalized to R events. The conditional probability $P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_{R-1})$ can be calculated by iteratively applying the previous procedure. The last step to obtain the distribution of $\mathbf{A}$ conditioned to all R events $\mathbf{B}_1, ..., \mathbf{B}_R$ is to establish the permanence of ratios relation:

$$\frac{\frac{1-P(\mathbf{A}|\mathbf{B}_1,...,\mathbf{B}_{R-1},\mathbf{B}_R)}{P(\mathbf{A}|\mathbf{B}_1,...,\mathbf{B}_{R-1},\mathbf{B}_R)}}{\frac{1-P(\mathbf{A}|\mathbf{B}_1,...,\mathbf{B}_{R-1})}{P(\mathbf{A}|\mathbf{B}_1,...,\mathbf{B}_{R-1})}} = \frac{\frac{1-P(\mathbf{A}|\mathbf{B}_R)}{P(\mathbf{A}|\mathbf{B}_R)}}{\frac{1-P(\mathbf{A})}{P(\mathbf{A})}}$$

This entails the general expression for the conditional probability under the permanence of ratios assumption:

$$P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_R) = \frac{\frac{1-P(\mathbf{A})}{P(\mathbf{A})}}{\frac{1-P(\mathbf{A})}{P(\mathbf{A})} + \frac{1-P(\mathbf{A}|\mathbf{B}_1,...,\mathbf{B}_{R-1})}{P(\mathbf{A}|\mathbf{B}_1,...,\mathbf{B}_{R-1})} \cdot \frac{1-P(\mathbf{A}|\mathbf{B}_R)}{P(\mathbf{A}|\mathbf{B}_R)}}$$

This expression does not require a prior knowledge of the relationships between the $\mathbf{B}_i$s, that is, all conditional relationships are built based on the assumption that the incremental information provided by the event $\mathbf{B}_i$ regarding the event $\mathbf{A}$ is constant regardless of the other conditioning events. The permanence of ratios assumption greatly simplifies the calculation of the conditional probability and it appears to correspond to a less restrictive assumption than the assumption of full independence between the events $\mathbf{B}_1, \ldots, \mathbf{B}_R$.

4.3.3 Multi-Gaussian Assumption

The redundancy between the sources of information can be calculated if the multivariate spatial distribution of the variable is known. This is not the case in general. The full multivariate distribution is known in the multi-Gaussian case.

Under the multi-Gaussian assumption, a multiple-point covariance can be fully described as combinations of second-order covariances (see for example, [134]). Simple indicator kriging can be used to estimate the conditional expectation of the indicator variable. A multiple-point event can be expressed as the set of all the single-point events that constitute it.

Although variables are not multi-Gaussian, this model can be used to approximate the redundancy term. In the context of integrating MP information, building an estimator that combines the estimate of the conditional probability at an unsampled location given different events could be considered. These events could be the indicator kriging (IK) estimate of the probability at $\mathbf{u}$, and MP probabilities for different configurations. These conditioning events are noted as $\mathbf{B}_1, \ldots, \mathbf{B}_R$. The conditional probability we are trying to estimate can be denoted as the event $\mathbf{A}$. Notice that because the assumption of multi-Gaussianity is used, the events must be disjoint, in the sense that no one single point should belong to more than one event. Otherwise, the simple indicator kriging system of equations will be singular.

The conditional probability of $\mathbf{A}$ can be written as a linear combination of the conditional probabilities calculated with each one of the conditioning events separately:

$$P(\mathbf{A}|\mathbf{B}_1, \ldots, \mathbf{B}_R) - P(\mathbf{A}) = \sum_{i=1}^{R} \omega_i \cdot (P(\mathbf{A}|\mathbf{B}_i) - P(\mathbf{A})) \tag{4.2}$$

If each event $\mathbf{B}_i$ is made of n_i single points, then, under the multi-Gaussian assumption, the joint distribution of all the events can be considered. The distribution is multivariate Gaussian of order $1 + \sum_{i=1}^{R} n_i$ with mean and variance-covariance matrix:

$$\mu_{(1+\sum_{i=1}^{R} n_i)\times 1} = \begin{pmatrix} P(\mathbf{A}) \\ \vdots \\ P(\mathbf{A}) \end{pmatrix}$$

$$\Sigma_{(1+\sum_{i=1}^{R} n_i)\times(1+\sum_{i=1}^{R} n_i)} = \begin{pmatrix} \Sigma_{A,A} & \Sigma_{A,B_1} & \cdots & \Sigma_{A,B_R} \\ \Sigma_{B_1,A} & \Sigma_{B_1,B_1} & \cdots & \Sigma_{B_1,B_R} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{B_R,B_1} & \Sigma_{B_R,B_2} & \cdots & \Sigma_{B_R,B_R} \end{pmatrix}$$

Each sub-matrix Σ_{B_i,B_j} of size $n_i \times n_j$ corresponds to the covariance matrix between the n_i single point events that constitute event $\mathbf{B}_i$ and the n_j single points that make the event $\mathbf{B}_j$:

$$\Sigma_{B_i,B_j} = \begin{pmatrix} Cov(\mathbf{u}_1^{B_i}, \mathbf{u}_1^{B_j}) & Cov(\mathbf{u}_1^{B_i}, \mathbf{u}_2^{B_j}) & \cdots & Cov(\mathbf{u}_1^{B_i}, \mathbf{u}_{n_j}^{B_j}) \\ Cov(\mathbf{u}_2^{B_i}, \mathbf{u}_1^{B_j}) & Cov(\mathbf{u}_2^{B_i}, \mathbf{u}_2^{B_j}) & \cdots & Cov(\mathbf{u}_2^{B_i}, \mathbf{u}_{n_j}^{B_j}) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(\mathbf{u}_{n_i}^{B_i}, \mathbf{u}_1^{B_j}) & Cov(\mathbf{u}_{n_i}^{B_i}, \mathbf{u}_2^{B_j}) & \cdots & Cov(\mathbf{u}_{n_i}^{B_i}, \mathbf{u}_{n_j}^{B_j}) \end{pmatrix}$$

When considering the events $\mathbf{B}_i$s as multiple-point events, the conditional probability of $\mathbf{A}$ given $\mathbf{B}_i$ is calculated as the simple indicator kriging estimate at location $\mathbf{u}$ given the n_i samples that constitute that multiple-point event.

The weights ω_i in **Equation 4.2** can be seen as a measure of closeness and redundancy between the event $\mathbf{B}_i$ and all the other events $\mathbf{B}_j$, with $j \neq i$.

Notice that the goal is to use a set of conditional probabilities $P(\mathbf{A}|\mathbf{B}_i), i = 1, \ldots, R$ obtained from different sources. The use of simple indicator kriging is meant to solve for the weights that yield the right conditional probability, if the variable was multivariate Gaussian and all the conditional probabilities were calculated by simple indicator kriging.

We calculate the conditional probability of **A** given a single event $\mathbf{B}_i$. This conditional probability can be seen as the indicator kriging estimate of **u**, given the vector $\mathbf{i}^{\mathbf{B}_i}$ of indicator values $i^{B_i}(\mathbf{u}_\alpha^{B_i}), \alpha = 1, ..., n_i$, or equivalently, as the linear regression estimate of the probability of **A** given $\mathbf{B}_i$:

$$P(\mathbf{A}|\mathbf{B}_i) - P(\mathbf{A}) = \Sigma_{A,B_i} \cdot \Sigma_{B_i,B_i}^{-1} \cdot (\mathbf{i}^{\mathbf{B}_i} - P(\mathbf{A})) \qquad \forall i = 1, ..., R \tag{4.3}$$

Now, expressing the conditional probability of **A** given all the events, that is, accounting for the redundancy between them, under the multi-Gaussian assumption:

$$\begin{aligned} P(\mathbf{A}|\mathbf{B}_1, ..., \mathbf{B}_R) - P(\mathbf{A}) &= \begin{pmatrix} \Sigma_{A,B_1} & \cdots & \Sigma_{A,B_R} \end{pmatrix} \cdot \\ & \begin{pmatrix} \Sigma_{B_1,B_1} & \cdots & \Sigma_{B_1,B_R} \\ \vdots & \ddots & \vdots \\ \Sigma_{B_R,B_1} & \cdots & \Sigma_{B_R,B_R} \end{pmatrix}^{-1} \cdot \begin{pmatrix} \mathbf{i}^{\mathbf{B}_1} - P(\mathbf{A}) \\ \vdots \\ \mathbf{i}^{\mathbf{B}_R} - P(\mathbf{A}) \end{pmatrix} \end{aligned}$$

To calculate the coefficient ω_i we can consider the event **C** being the combination of all the events $\mathbf{B}_j, j \neq i$. The expression for the conditional probability can be rewritten as:

$$P(\mathbf{A}|\mathbf{B}_i, \mathbf{C}) - P(\mathbf{A}) = \begin{pmatrix} \Sigma_{A,B_i} & \Sigma_{A,C} \end{pmatrix} \cdot \begin{pmatrix} \Sigma_{B_i,B_i} & \Sigma_{B_i,C} \\ \Sigma_{C,B_i} & \Sigma_{C,C} \end{pmatrix}^{-1} \cdot \begin{pmatrix} \mathbf{i}^{\mathbf{B}_i} - P(\mathbf{A}) \\ \mathbf{i}^{\mathbf{C}} - P(\mathbf{A}) \end{pmatrix}$$

The inverse of the covariance matrix can be written as:

$$\begin{pmatrix} \Sigma_{B_i,B_i} & \Sigma_{B_i,C} \\ \Sigma_{C,B_i} & \Sigma_{C,C} \end{pmatrix}^{-1} = \begin{pmatrix} D_1 & D_0 \\ D_0^T & D_2 \end{pmatrix}$$

with

$$\begin{aligned} D_1 &= (\Sigma_{B_i,B_i} - \Sigma_{B_i,C} \cdot \Sigma_{C,C}^{-1} \cdot \Sigma_{C,B_i})^{-1} \\ D_0 &= -(\Sigma_{B_i,B_i} - \Sigma_{B_i,C} \cdot \Sigma_{C,C}^{-1} \cdot \Sigma_{C,B_i})^{-1} \cdot \Sigma_{B_i,C} \cdot \Sigma_{C,C}^{-1} \\ D_2 &= \Sigma_{C,C}^{-1} + \left((\Sigma_{B_i,B_i} - \Sigma_{B_i,C} \cdot \Sigma_{C,C}^{-1} \cdot \Sigma_{C,B_i})^{-1} \cdot \Sigma_{B_i,C} \cdot \Sigma_{C,C}^{-1}\right)^T \cdot \Sigma_{B_i,C} \cdot \Sigma_{C,C}^{-1} \end{aligned}$$

That is,

$$\begin{aligned} P(\mathbf{A}|\mathbf{B}_i, \mathbf{C}) - P(\mathbf{A}) &= (\Sigma_{A,B_i} \cdot D_1 + \Sigma_{A,C} \cdot D_0^T) \cdot (\mathbf{i}^{\mathbf{B}_i} - P(\mathbf{A})) + \\ & (\Sigma_{A,B_i} \cdot D_0 + \Sigma_{A,C} \cdot D_2) \cdot (\mathbf{i}^{\mathbf{C}} - P(\mathbf{A})) \end{aligned} \tag{4.4}$$

In order to determine the weight assigned to the conditional probability $P(\mathbf{A}|\mathbf{B}_i)$, we can rewrite the conditional probability on **Equation 4.2** as:

$$P(\mathbf{A}|\mathbf{B}_i, \mathbf{C}) - P(\mathbf{A}) = \omega_i \cdot (P(\mathbf{A}|\mathbf{B}_i) - P(\mathbf{A})) + \omega_C \cdot (P(\mathbf{A}|\mathbf{C}) - P(\mathbf{A}))$$

and using **Equation 4.3**, this can be written as:

$$
\begin{aligned}
P(\mathbf{A}|\mathbf{B}_i, \mathbf{C}) - P(\mathbf{A}) \quad = \quad & \omega_i \cdot \left(\Sigma_{A,B_i} \cdot \Sigma_{B_i,B_i}^{-1} \cdot (\mathbf{i}^{\mathbf{B}_i} - P(\mathbf{A}))\right) + \\
& \omega_C \cdot \left(\Sigma_{A,C} \cdot \Sigma_{C,C}^{-1} \cdot (\mathbf{i}^{\mathbf{C}} - P(\mathbf{A}))\right)
\end{aligned} \tag{4.5}
$$

To identify the conditional probabilities of **Equations 4.4** and **4.5**, the approximate values for ω_i and ω_C can be defined:

$$
\omega_i = \frac{\|\Sigma_{A,B_i} \cdot D_1 + \Sigma_{A,C} \cdot D_0^T\|}{\|\Sigma_{A,B_i} \cdot \Sigma_{B_i,B_i}^{-1}\|} \qquad \omega_C = \frac{\|\Sigma_{A,B_i} \cdot D_0 + \Sigma_{A,C} \cdot D_2\|}{\|\Sigma_{A,C} \cdot \Sigma_{C,C}^{-1}\|} \tag{4.6}
$$

Because our goal is to combine two estimates, the weights must be numbers, not vectors. The ideal case to apply this method is when the vector $\Sigma_{A,B_i} \cdot D_1 + \Sigma_{A,C} \cdot D_0^T$ is proportional to $\Sigma_{A,B_i} \cdot \Sigma_{B_i,B_i}^{-1}$, and the vector $\Sigma_{A,B_i} \cdot D_0 + \Sigma_{A,C} \cdot D_2$ is proportional to $\Sigma_{A,C} \cdot \Sigma_{C,C}^{-1}$. In general, these pairs of vectors are not proportional to each other. The calculation of ω_i and ω_C under these circumstances is an approximation of the proportionality coefficient if they were proportional.

We can consider some extreme cases of the linear combination of estimates presented above. A first extreme case is to assume independence between the events. This amounts to setting the cross-terms $\Sigma_{B_i,C}$ and Σ_{C,B_i} as 0. D_0 becomes zero, D_1 is reduced to $(\Sigma_{B_i,B_i})^{-1}$, and D_2 is simplified to $(\Sigma_{C,C})^{-1}$. Thus, $\omega_i = \omega_C = 1$. The estimate built as a linear combination is then just the sum of the two (independent) estimates of the conditional probability.

Independence of $\mathbf{B}_i$ and $\mathbf{C}$ is, in the context presented here, not a good assumption, since $\mathbf{C}$ is very close to $\mathbf{u}$ and $\mathbf{B}_i$ is correlated with $\mathbf{A}$, therefore it should also be correlated with $\mathbf{C}$.

Notice that this procedure to estimate the weights that measure the redundancy between the sources of information can be extended to an arbitrary number of conditioning events R. The theory has been presented looking at the quantification of the redundancy for a single event given all others ($\mathbf{C}$), however the same reasoning can be applied to define simultaneously the weights for multiple conditioning events.

In a general case, where the indicator variable does not come from a multivariate Gaussian distribution but where quantifying the redundancy between the two sources of information $\mathbf{B}_i$ and $\mathbf{C}$ appears not possible, the weighting proposed under the multi-Gaussian assumption could be used to improve the solution, although it would be just an approximation.

4.3.4 Comments

In the implementation of these methods a few problems may arise.

First, the property of Bayesian updating under the full independence assumption between the sources of information can lead to severe order relation deviations, since under this assumption, the updated probability can easily take values above one. The more complex the implementation, that is, if multiple sources are used and deemed independent from one another, the problem will worsen making its practical use inadequate.

This problem is not encountered under the assumption of permanence of ratios and under the multi-Gaussian assumption to assess the relationship between the sources of information. Permanence of ratios gives an updated probability always

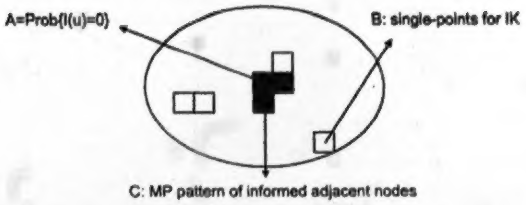


Figure 4.1: Example of three events. **A** corresponds to the probability of the grey node not to exceed a threshold value. Six single-point events are found in a search neighborhood of the node being estimated. They are illustrated as white and black nodes. **B** is formed by all the single point events that are not part of the multiple-point configuration that constitutes **C**. From the six single-points found in the search neighborhood, only the white nodes ($n_B = 4$ points) will be used to get the IK estimate, to avoid singularity due to full redundancy with the nodes in **C**. **C** is the multiple-point event formed by $n_C = 2$ points adjacent to the node whose conditional probability is being estimated. They appear as black nodes. The probability of **A** given the information in **C** is obtained either from a training image or from data.

within the interval [0,1], provided each component falls within this interval. In the case of approximating the redundancy through a multi-Gaussian assumption, order relations should be minor, since the weighting will prevent the estimated probability to fall outside [0,1]. Deviations should occur with approximately the same frequency as when IK is used.

A second problem that becomes evident when these methods are applied is the inconsistency between the univariate distributions of the different sources of information. The methods assume that the proportions below the thresholds are the same no matter what source of information is considered. This is not always the case. Corrections are needed to solve this problem. The goal is to make the estimator unbiased. The simplest solution is to replace the term for $P(\mathbf{A})$ by the probability calculated from the source that differs from the target probability.

4.4 Practical Implementation

The methods for integrating different events have been implemented. Consider the following definitions for the events (**Figure 4.1**):

A : The event of having $Z(\mathbf{u}) \leq z_k$. z_k corresponds to a threshold value.

B : The n_B single points used to inform the location **u**. These points are located in a neighborhood of **u**. They correspond to the data used to solve the simple indicator kriging system to estimate the conditional probability at **u**.

C : The multiple-point event formed by a set of n_C points, usually very close or adjacent to the location of interest **u**.

B will help reproduce the long range on the variable, while **C** will locally modify the estimation of the probabilities to consider more complex multiple-point information. It will integrate some of the MP information to the IK estimator.

Bayes' postulate says that the conditional probability of having the event **A** given the information **B**, and **C** is given by:

$$P(\mathbf{A}|\mathbf{B},\mathbf{C}) = \frac{P(\mathbf{A},\mathbf{B},\mathbf{C})}{P(\mathbf{B},\mathbf{C})} = \frac{P(\mathbf{C}|\mathbf{A},\mathbf{B}) \cdot P(\mathbf{B}|\mathbf{A}) \cdot P(\mathbf{A})}{P(\mathbf{B},\mathbf{C})}$$

This requires the knowledge of the joint probability between **B** and **C**. Often this joint probability is hard to infer, in particular when the data used in **B** and **C** are different variables and/or at different supports. In our application, we would need to infer the covariance between n_B single point events and the MP event made of n_C points. This is difficult, thus the idea is to find some way to avoid these calculations.

The inference of **C** is not difficult: there is no need to model the multiple-point statistics to ensure positive definiteness, although if these probabilities are not consistent with the ones obtained for **A** given **B**, order relations deviations will likely increase. **B** is obtained by simple indicator kriging, thus the standard practice of calculation, interpretation and modelling of the indicator variograms is required.

C is calculated for a given set of multiple-point patterns according to the configuration of the data. In case of having a large training image, very complicated patterns could be used to condition the estimation. If only data are available, a limited number of restrictive configurations should suffice to improve the final numerical model.

In the following examples, a set of very simple two-dimensional patterns are used and the probabilities are inferred from training images built to investigate the performance of each technique. The idea is to use only the four adjacent nodes to the one being estimated (**Figure 4.2**). The probabilities are associated with the frequencies of having those configurations in the training image. Since every node is coded as an indicator, for each p-point configuration there are 2^p combinations of zeros and ones possible. The total number of MP events from which the probabilities of occurrence must be obtained, is 81:

$$\sum_{i=0}^{4} \binom{i}{4} \times 2^i = 1 \times 1 + 4 \times 2 + 6 \times 4 + 4 \times 8 + 1 \times 16 = 81$$

If a combination of the indicators is not found in the training image, no updating will take place and the updated probability will correspond exactly to the one obtained by indicator kriging.

As mentioned before, this idea can be extended to any configurations that fit the data or training image. For example, the use of a linear vertical pattern would match the spatial configuration of drillhole data or well data (**Figure 4.3**). With these patterns, the total number of probabilities needed is 41.

4.4.1 Sequential Multiple-Point Simulation

The probability of a point being below a threshold given some multiple-point configuration can be extracted from a training image, or even from data. The idea of a sequential MP simulation is then to visit randomly the nodes in the model and check for the nodes within a specific pattern. In all the examples presented here, the pattern is made of the 4 adjacent nodes to the node of interest. These can be

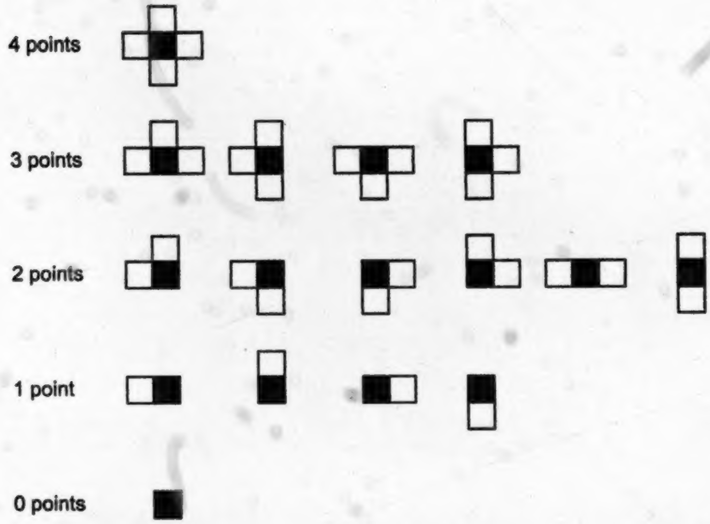

4 points
3 points
2 points
1 point
0 points

Figure 4.2: Multiple-point patterns with adjacent grid nodes. The gray node is the one being estimated. The patterns correspond to the four adjacent nodes to the node of interest. The probabilities are extracted from the training image even when some of the nodes are not informed, generating the three-, two-, one-, and zero-point patterns.

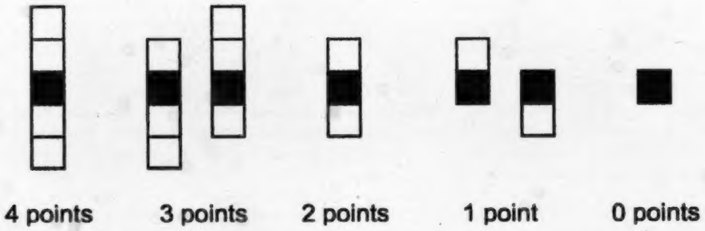

4 points
3 points
2 points
1 point
0 points

Figure 4.3: Multiple-point patterns extracted from drillhole or well data. Only connected patterns are considered.

data values or previously simulated nodes. These MP probabilities correspond to $P(\mathbf{A}|\mathbf{C})$.

4.5 Applications

Several reference images have been built to have a training image for MP statistics, and for a visual reference. Five methods are used: sequential indicator simulation (`SIS`), MP simulation, updating under the independence assumption between **B** and **C**, updating under the permanence of ratios assumption, and combining estimates under the multi-Gaussian assumption. The five methods are simulated unconditionally and with the same random path for comparison.

4.5.1 Binary Examples

Binary images are built with the object based algorithm `ellipsim` in `GSLIB` [39]. Different proportions are tested (p= 10, 50, and 90%) and cases with and without anisotropy.

This section also presents some binary examples using training images of continuous variables.

Small Isotropic Objects

Objects of radius equal to 3 units on a two-dimensional domain of 100 by 100 nodes, with a spacing of one unit are simulated as a reference. The five methods proposed are used to generate one realization of the phenomenon. **Figures 4.4**, **4.5**, and **4.6** show the results.

Large Isotropic Objects

Larger objects were simulated, with a radius of 9 units, under the same condition as before. The results are shown on **Figures 4.7**, **4.8**, and **4.9**.

Small Anisotropic Objects

Ellipses with an anisotropy at 30 degrees and major radius of 6 units and minor radius of 3 units were generated, again using different proportions (10, 50, and 90 %). The results are presented in **Figures 4.10**, **4.11**, and **4.12**.

Large Anisotropic Objects

Finally, larger anisotropic ellipses were generated to extract the MP statistics and then simulated with the five methods. The major radius is 18 and the minor radius is 9 units. **Figures 4.13**, **4.14**, and **4.15** show the resulting maps.

MP Statistics Extracted From a Continuous Training Image

A training image showing a continuous variable is used in this example. The median threshold has been chosen to create a binary image. The probabilities of multiple-point events as shown on **Figure 4.2** are extracted to update the `SIS` algorithm. The

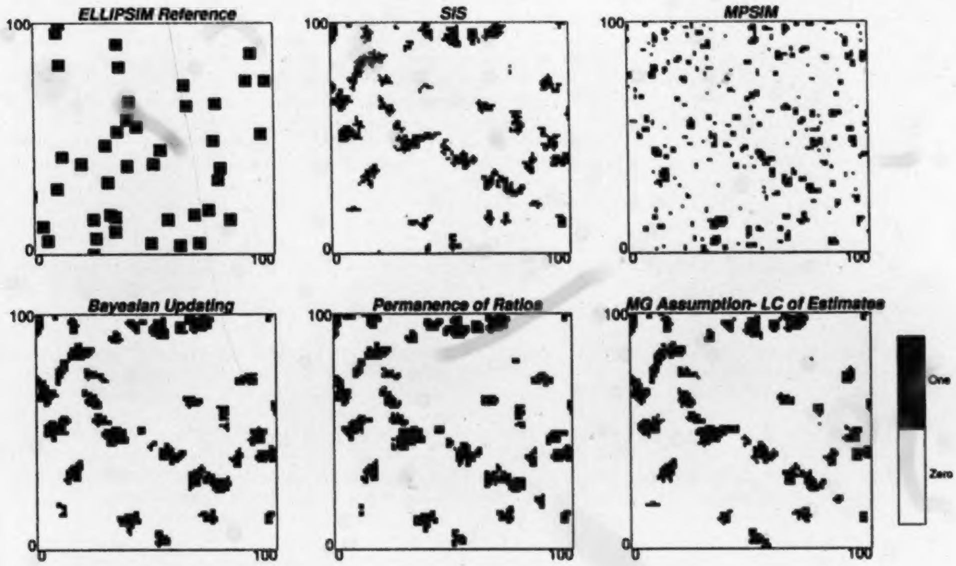


Figure 4.4: Maps of simulated values for small isotropic objects. Proportion above the threshold is 10 %

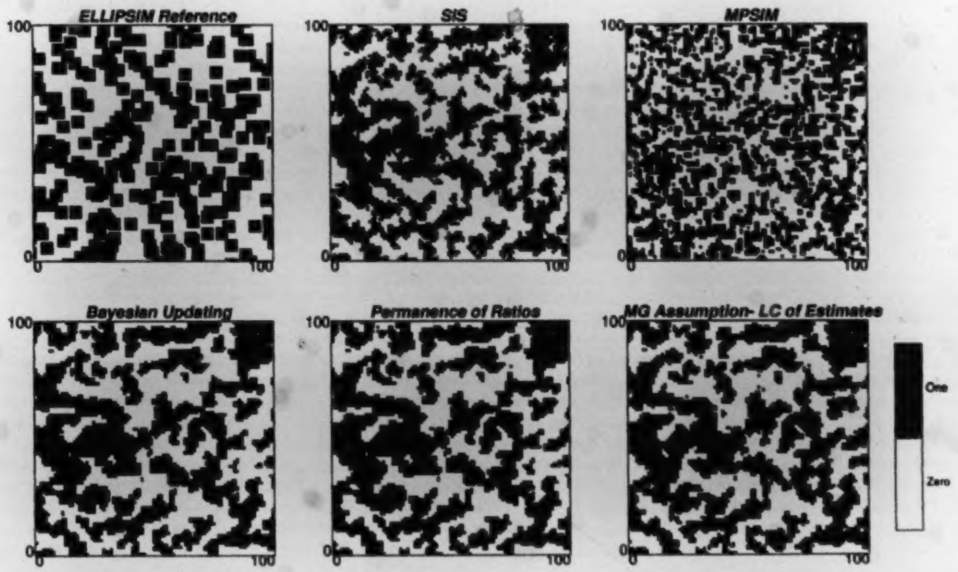


Figure 4.5: Maps of simulated values for small isotropic objects. Proportion above the threshold is 50 %

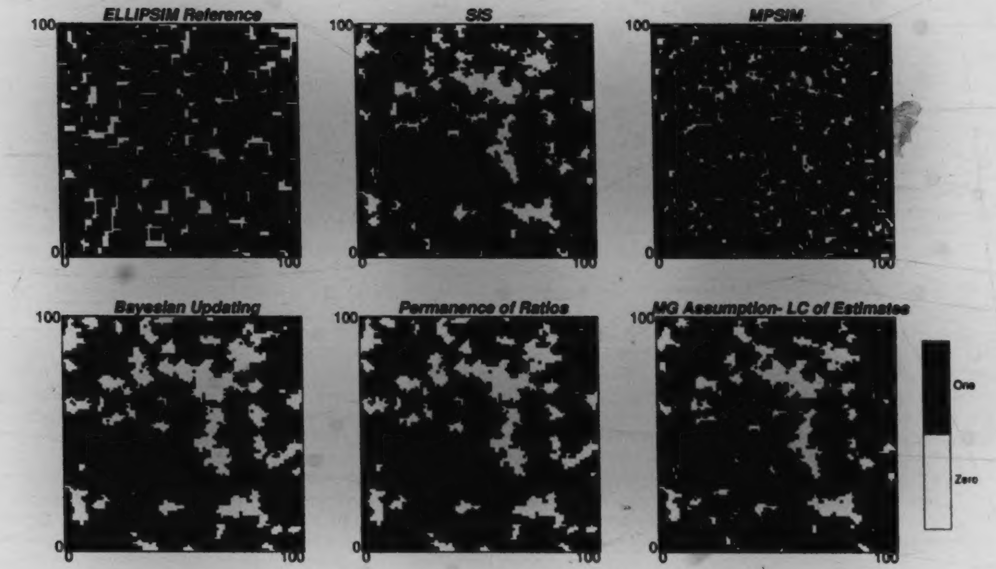


Figure 4.6: Maps of simulated values for small isotropic objects. Proportion above the threshold is 90 %

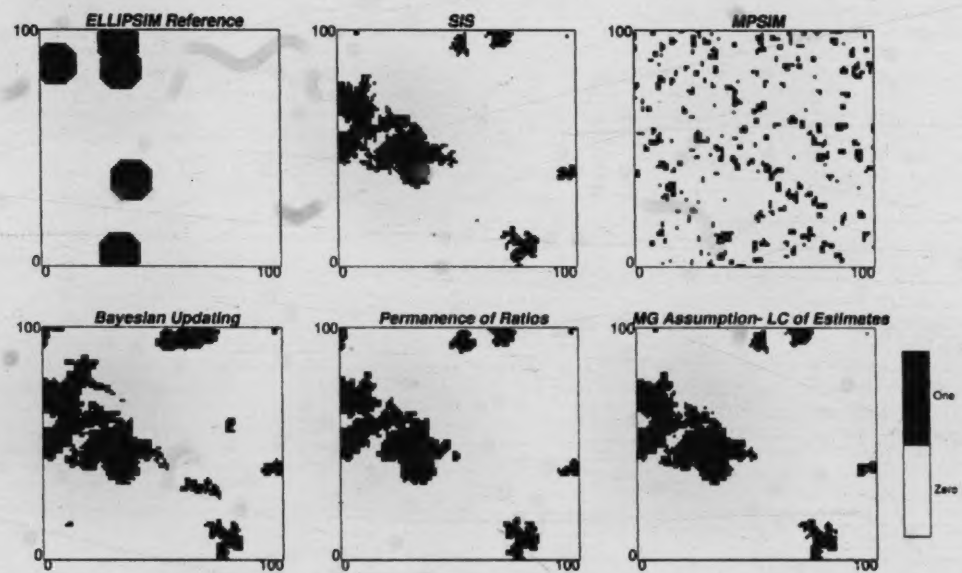


Figure 4.7: Maps of simulated values for large isotropic objects. Proportion above the threshold is 10 %

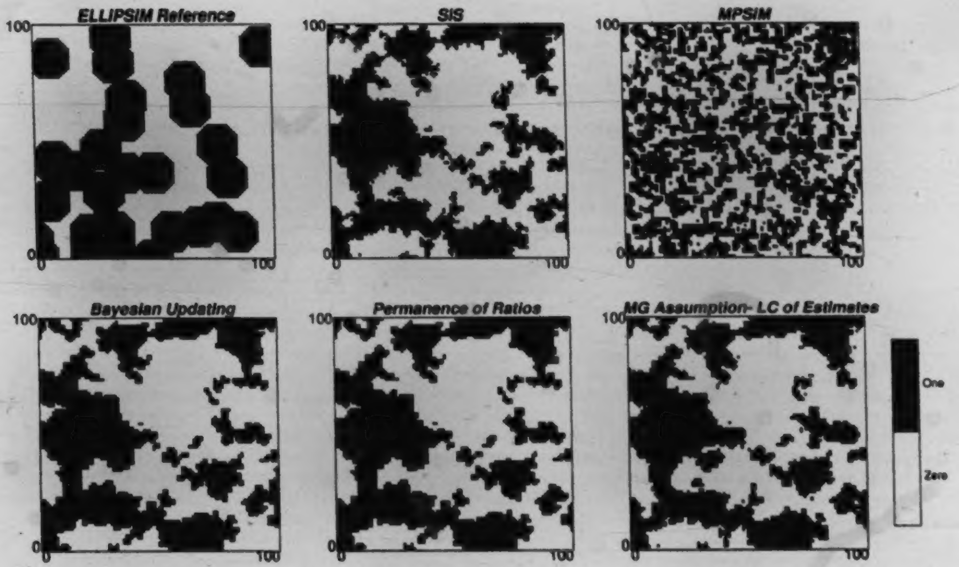

ELLIPSIM Reference
SIS
MPSIM
Bayesian Updating
Permanence of Ratios
MG Assumption- LC of Estimates
100
0
100
One
Zero

Figure 4.8: Maps of simulated values for large isotropic objects. Proportion above the threshold is 50 %

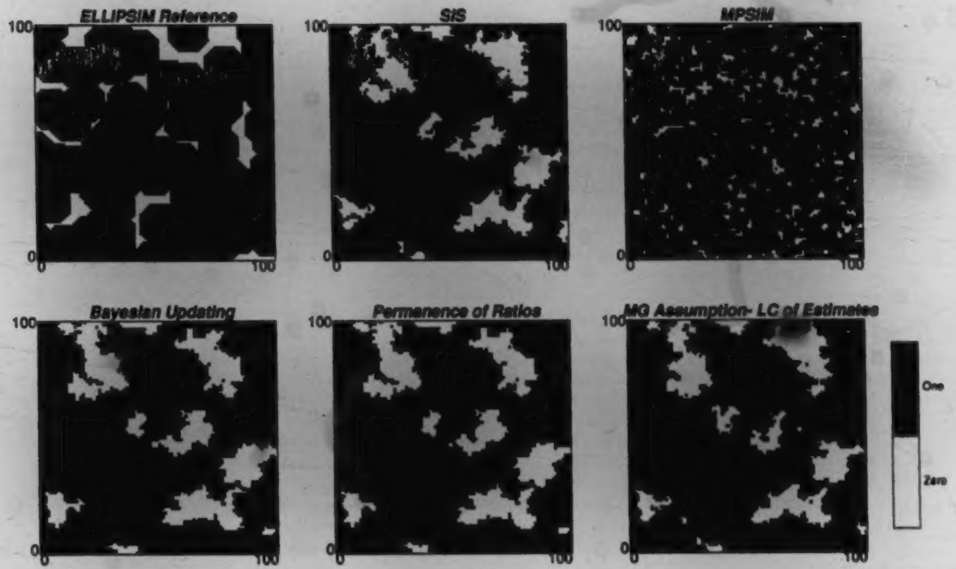

ELLIPSIM Reference
SIS
MPSIM
Bayesian Updating
Permanence of Ratios
MG Assumption- LC of Estimates
100
0
100
One
Zero

Figure 4.9: Maps of simulated values for large isotropic objects. Proportion above the threshold is 90 %

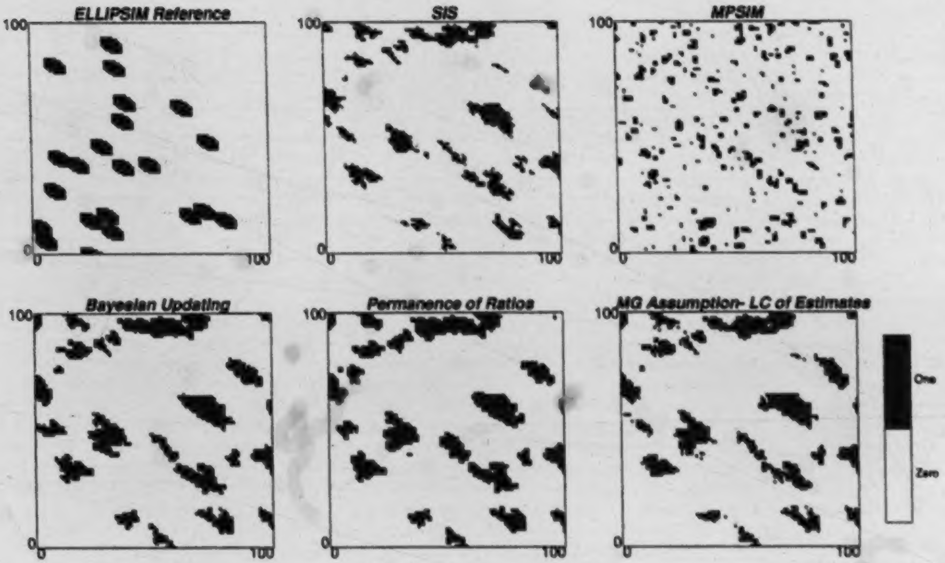

ELLIPSIM Reference
SIS
MPSIM
Bayesian Updating
Permanence of Ratios
MG Assumption- LC of Estimates
One
Zero
100
0
100

Figure 4.10: Maps of simulated values for small anisotropic objects. Proportion above the threshold is 10 %

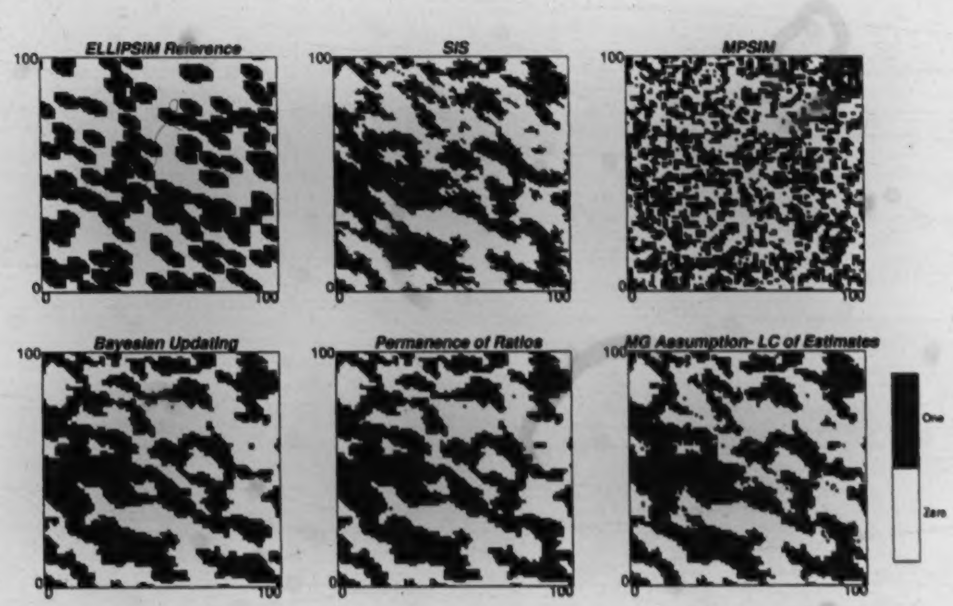

ELLIPSIM Reference
SIS
MPSIM
Bayesian Updating
Permanence of Ratios
MG Assumption- LC of Estimates
One
Zero
100
0
100

Figure 4.11: Maps of simulated values for small anisotropic objects. Proportion above the threshold is 50 %

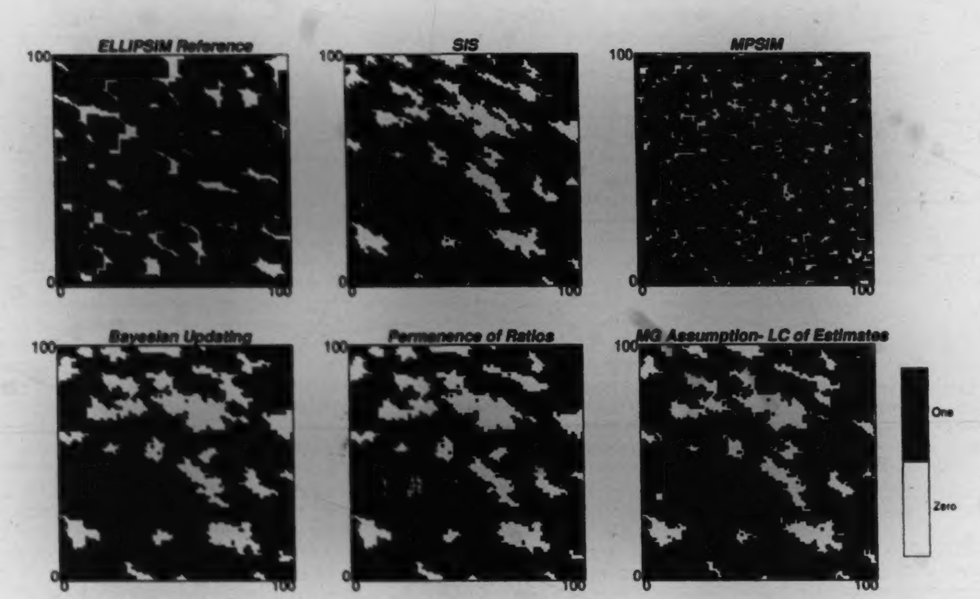

ELLIPSIM Reference
SIS
MPSIM
Bayesian Updating
Permanence of Ratios
MG Assumption- LC of Estimates
100
0
100
One
Zero

Figure 4.12: Maps of simulated values for small anisotropic objects. Proportion above the threshold is 90 %

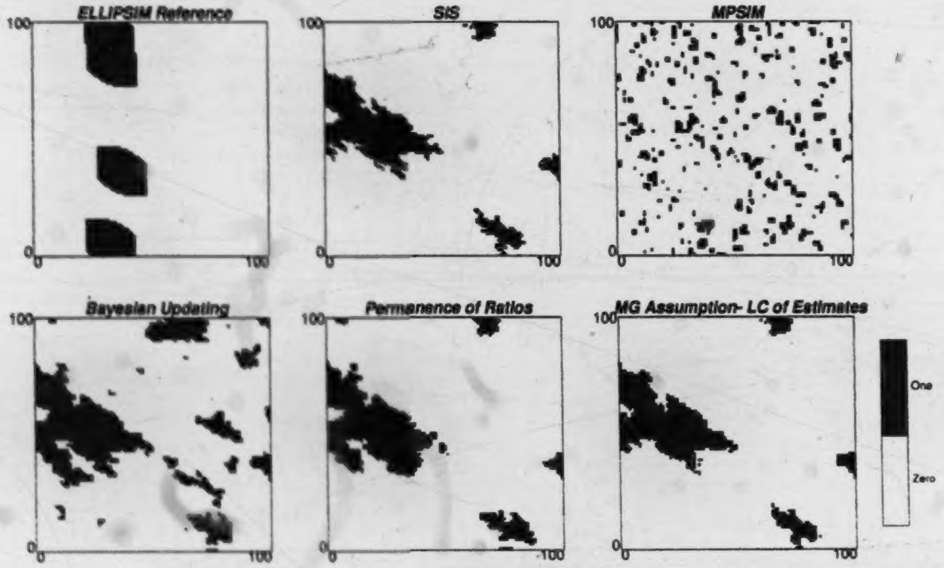

ELLIPSIM Reference
SIS
MPSIM
Bayesian Updating
Permanence of Ratios
MG Assumption- LC of Estimates
100
0
100
One
Zero

Figure 4.13: Maps of simulated values for large anisotropic objects. Proportion above the threshold is 10 %

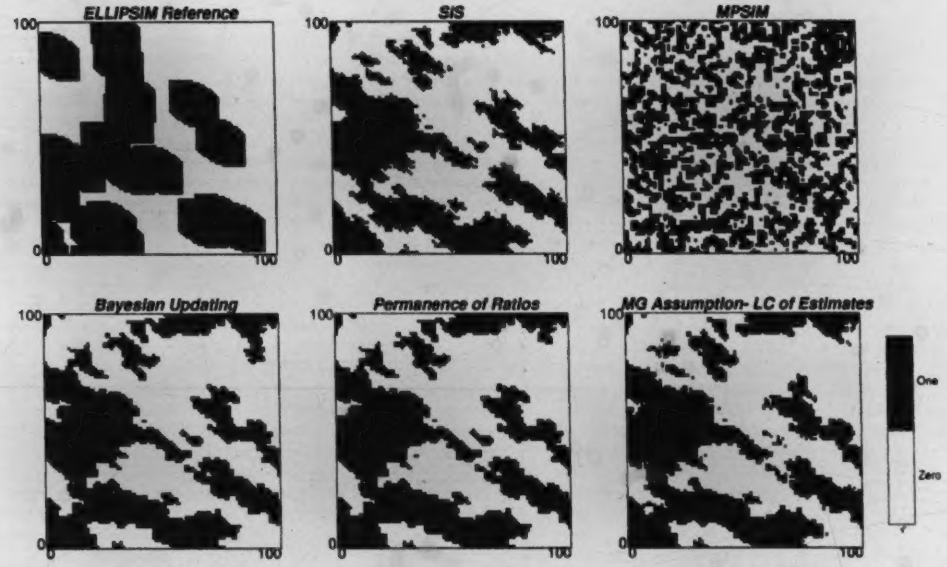


Figure 4.14: Maps of simulated values for large anisotropic objects. Proportion above the threshold is 50 %

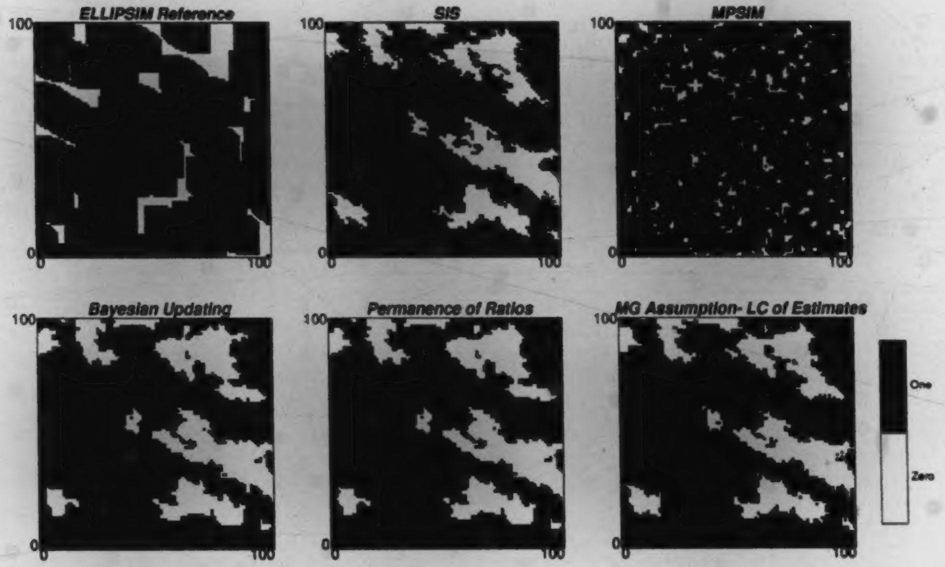


Figure 4.15: Maps of simulated values for large anisotropic objects. Proportion above the threshold is 90 %

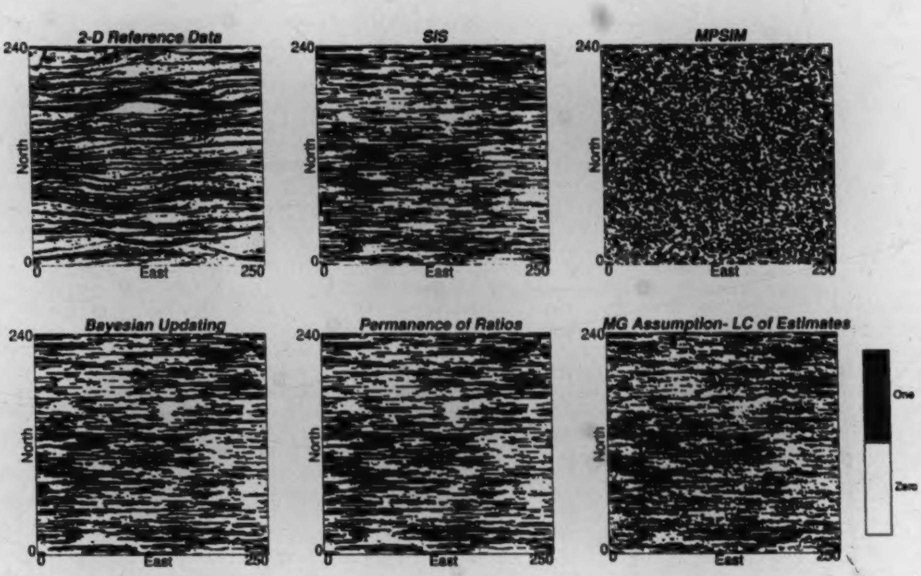


Figure 4.16: Maps of simulated values for a binary image taken from a continuous variable. Proportion above the threshold is 50 %

results are shown on **Figure 4.16**. Although the curvilinearity is not captured by any of the methods, the layering that can be seen on the training image is also seen on the simulated models. The models obtained with Bayesian updating assuming data independence and permanence of ratios look too clean. These two methods erase all the noise that can be seen in the reference. On the contrary, the updating by combining the estimates of the conditional probability under the multi-Gaussian assumption seems to add too much noise.

4.5.2 Continuous Variable Example

The same training image of a continuous variable used in the last example is now utilized to simulate using multiple thresholds. A variable with a positively biased distribution has been characterized with 10 thresholds corresponding to each one of the 9 deciles, in addition to the quantile 0.95. The indicator variograms for each thresholds were calculated to perform sequential indicator simulation. The frequency of occurrence of the MP-patterns that have at least one adjacent node informed are extracted from the reference image. These statistics are then used to update the `SIS` simulated model. Again, all five methods were applied and the results are shown on **Figure 4.17**.

4.5.3 Discussion

There are several comments:

- `SIS` generates a map with the correct long range continuity, but in the short range there seems to be too much noise: multiple-point statistics are clearly not

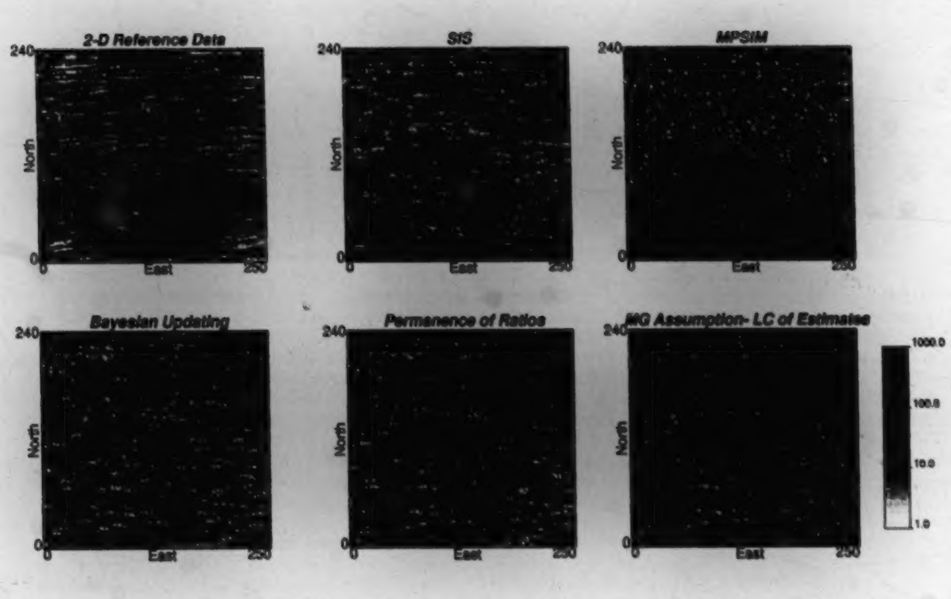


Figure 4.17: Maps of the continuous simulated values using all five methods for a continuous variable. Ten thresholds were used to characterize the spatial continuity of the variable.

reproduced. This problem is partially solved via image cleaning. Algorithms such as MAPS reduce the excessive randomness, but without direct control of the multivariate distribution.

- MP simulation by itself does not correct for this problem due to the short range of the pattern and the random path used. When there are no adjacent nodes informed, the algorithm will simply draw from the global, generating a very noisy image. The long-range structure is not captured by this algorithm. Some attempts were made to correct this problem, such as using a regular path, however the resulting models showed artifacts of the path chosen.

- Updating under the data independence assumption cleans the image. It looks like an SIS output cleaned, for example, with a MAPS algorithm (maximum a posteriori selection) [40]. The short-range anisotropic features of the SIS output look more isotropic, maybe because of the small size of the pattern used to extract the MP statistics, which does not reflects anisotropies very clearly. In general, the result is more similar to the training image.

- Updating under the permanence of ratios assumption also cleans the image. It is hard to judge which method gives the better result.

- The idea of combining both estimates of the conditional probability, the one calculated by indicator kriging of the data not belonging to the pattern used to infer the MP statistics, and the probability obtained from the MP pattern, performed surprisingly well. It compares similarly to the algorithm where the updating is made under the data independence and permanence of ratios assumptions. However, it approximates the effect of redundancy between both sources of information, so this assumption should be more realistic.

The examples presented correspond to unconditional simulations. It was found in these examples that all the methods generated a bias on the probabilities below each threshold. This bias was always towards the median, that is, if the global proportion below a threshold was 0.10, the resulting proportion in the simulated model was 0.15. Similarly, if the target proportion was 0.90, the resulting proportion in the simulated model was 0.85. Interestingly, if the threshold corresponded to the median, no bias was found.

The fact that sequential indicator simulation departs from the target proportions even in the case of having conditioning data, for categorical variables is known and a correction for that case has already been proposed [69, 98, 151].

Investigation of the bias in our case of simulating continuous variables unconditionally showed that it is due to the correction of order relation deviations.

This also happens when sequential indicator simulation is implemented without incorporating MP statistics. For the cutoffs corresponding to the tenth and ninetieth percentiles, the bias is around 2.5 % towards the median. For the fiftieth percentile, no bias was found.

The bias can be explained by recalling that the IK estimate of the probability to be below a threshold is an unbiased estimator (recall **Equation 2.6**). However, the estimated value may lie outside the allowed interval for a probability, that is, outside [0,1]. Thus, a correction is required (see **Section 2.3.6**).

Considering a binary simulation, that is, when only one threshold is being used, say the ninetieth percentile, the bias is introduced by correcting more often deviations due to having an estimate greater than one, than deviations due to the estimate being less than zero. Overall, the estimated values are no longer unbiased.

Figure 4.18 shows the histogram of corrections required during a run of sequential indicator simulation with a threshold at the percentile 90. Overall, corrections are biased, giving a non-zero average. Furthermore, when inspecting the histograms of positive and negative corrections, two facts are evident: first, positive corrections are made in more than 95% of the cases where a correction is required; and second, the average of the positive corrections is much smaller than the average of the negative corrections. However, negative corrections are fewer than positive corrections, and despite their larger magnitude, they are not enough to counterbalance the positive corrections, leaving an overall positive bias in the estimation of the probabilities. A similar problem can be seen when considering different thresholds. When the median is used, the corrections for values above one and below zero are similar, cancelling each other and generating no bias.

One way to overcome this problem is to dynamically correct for the bias introduced, every time a correction is made. This has been implemented with favorable results. The idea is to keep track of the last order relation correction made, and to adjust the next estimate by that amount, in order to generate overall unbiased realizations.

This dynamic correction generates a slight increase in the nugget effect, which is seen as a shift in the experimental variogram of the realization. The change is not significant if the corrections are small (**Figure 4.19**).

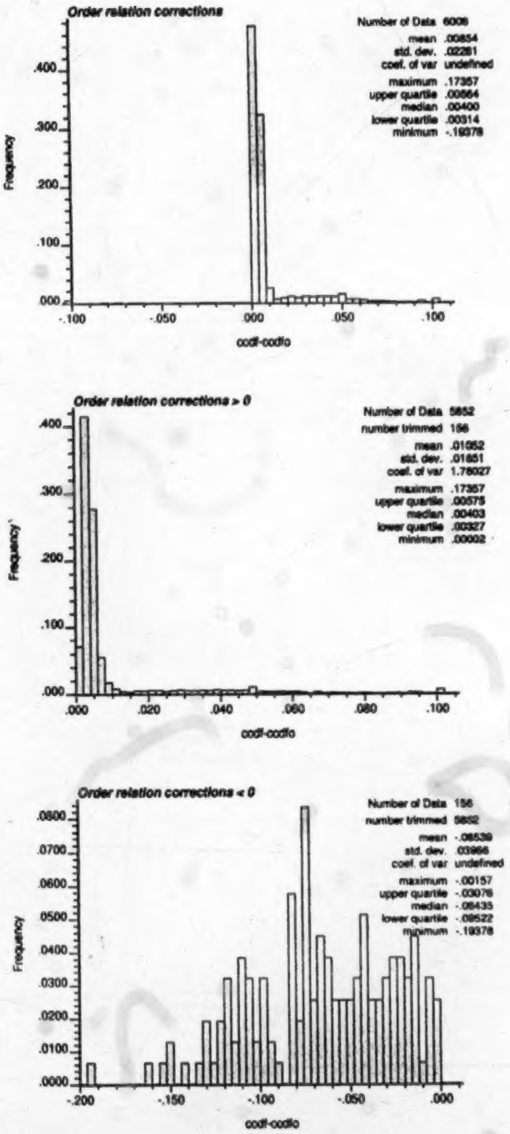


Figure 4.18: Histograms of order relation corrections in SIS. The top histogram shows all corrections together, the middle one shows the negative corrections and the bottom one shows the positive order relation corrections.

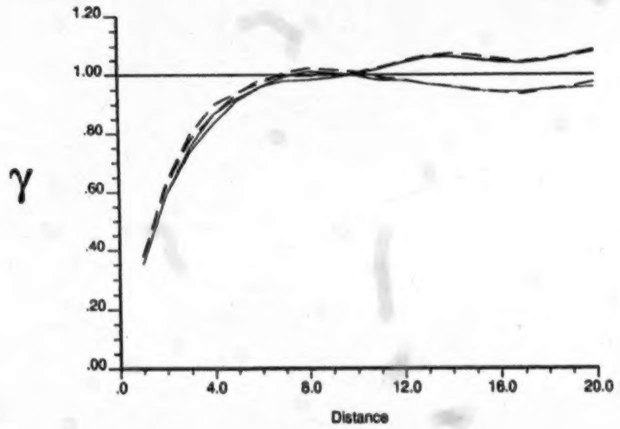


Figure 4.19: Experimental indicator variogram before (continuous line) and after dynamically correcting for the bias introduced by order relation corrections (dashed line) in `SIS`.

The same problem occurs with the proposed methods for updating the IK probabilities with MP statistics. The magnitude of the order relation corrections will dictate if a dynamic correction such as the one applied for `SIS` gives satisfactory results. However, it is known that `SIS` performs well without that correction, if the following conditions are met:

- Enough conditioning information is available.
- The size of the simulated domain is large with respect to the range of correlation of the variogram.
- Multiple grid search is used to simulate.

We expect that the proposed updating methods will also perform well under these circumstances. A problem with the updating of the IK probabilities with multiple-point statistics under the assumption of independence of these probabilities is foreseen, since the order relation deviations for this updating technique are extremely large.

4.6 Assessing Performance

Comparing the methods is not an easy task. One straightforward approach would be to look at the histograms of MP probabilities obtained from each simulated model, compared with the reference probabilities. However, there are 81 geometrical configurations and a measure of closeness to the truth is not simple to measure, since certain small deviations could be important. Classical measures of mismatch could be used, such as a mean squared error or mean absolute error. Unfortunately, not all the MP configurations have the same importance, so a small mismatch in an important configuration can pass unnoticed with a summary statistic of this kind.

Algorithm	SIS	Full Indep.	Perm. of Ratios	MG Assumption
MSE	3.18E-05	1.98E-05	2.48E-05	4.78E-06
MAS	3.36E-02	3.85E-02	4.14E-02	2.71E-02

Table 4.1: Mean squared error and mean absolute error in the MP probability for the different algorithms.

This analysis was done to the binary example generated with a training image from a continuous variable; otherwise, we would need to graph the probabilities for each threshold and cross-probabilities between thresholds.

The probabilities of each one of the 81 MP configurations was calculated for the reference and for each one of the 5 maps generated with the different algorithms. In all rigor, this should be done over multiple realizations with each algorithm, to avoid problems of ergodicity. The mismatch between the MP probability for each configuration from the training image and from each simulated map was calculated and plotted (**Figure 4.20**). The model resulting under the permanence of ratios assumptions appears visually as the best, however small deviations from the target MP probabilities may have a large impact on response of the model after a transfer function. The absolute value of this error is also presented in **Figure 4.21**. In this plot, the multi-Gaussian assumption appears as the closest to the target probabilities. The results for the sequential MP simulation approach are not presented, since the models generated with this algorithm did not share the long range correlation required.

Interestingly, the graphs show quite clearly that the estimation of the conditional probability combining the IK estimate (from the data) and the MP probability estimate (from the training image) generates in general smaller errors than the other algorithms. This is also seen when looking at summary statistics, such as the mean squared error and the mean absolute error (**Table 4.1**). Cases where a given configuration of the multiple-points was not found are not easily comparable, since the impact of not having a configuration that is present in the target statistics cannot easily be quantified.

Implementation with real data should provide more insight regarding which method performs the best. The exploratory examples developed in this section only help to anticipate the improvement in performance of each one of the methodologies proposed.

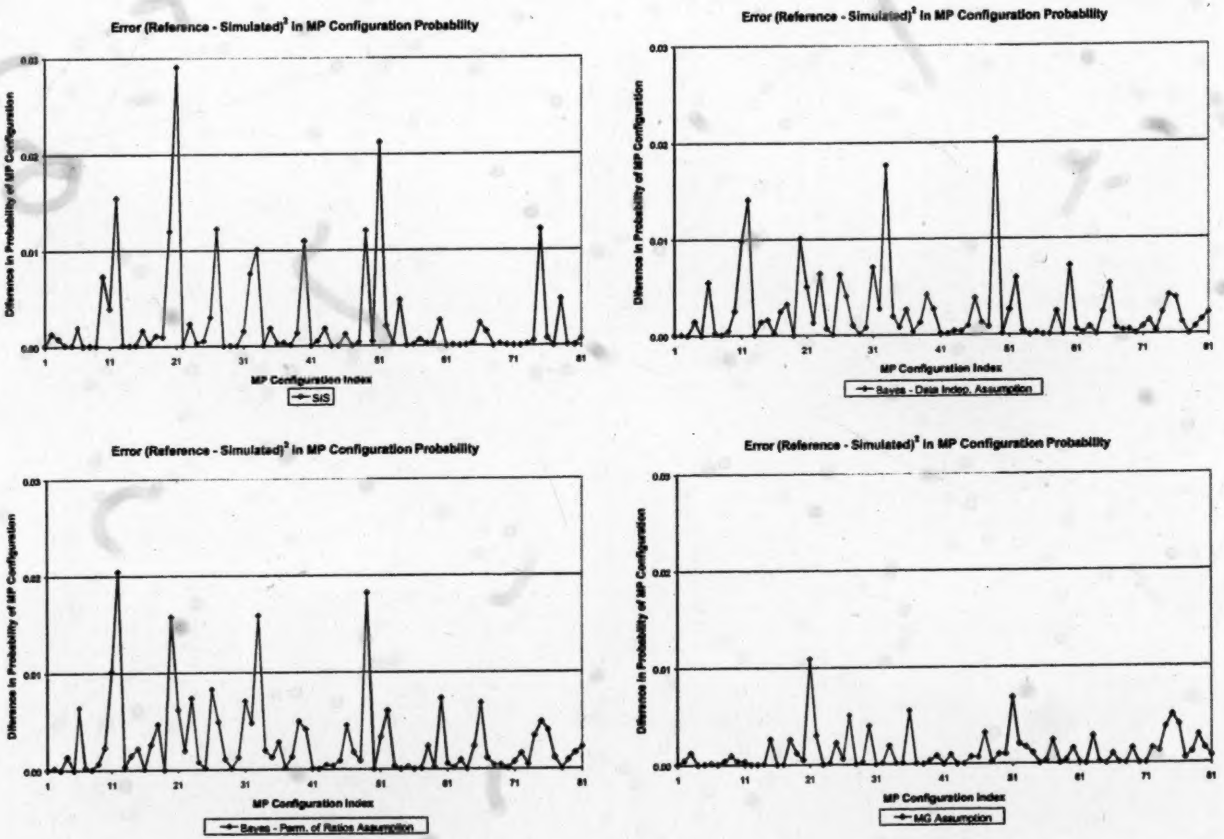


Figure 4.20: Mismatch in MP probability for all 81 MP configurations for four of the methods.

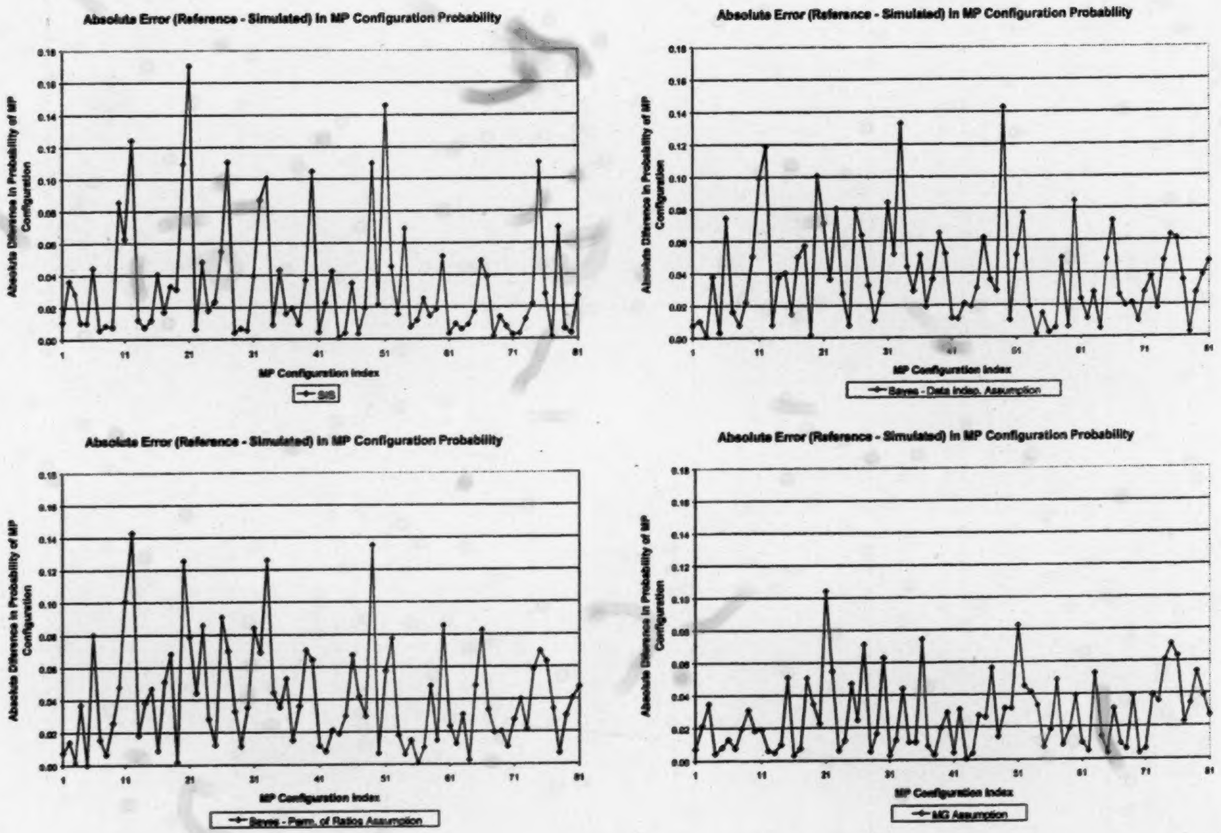


Figure 4.21: Absolute value of the mismatch in MP probability for all 81 MP configurations for four of the methods.

Example	Full Indep.	Perm. of Ratios	MG Assumption
Small isotropic objects p = 0.10	0.76	0.25	0.65
Small isotropic objects p = 0.50	0.66	0.45	0.55
Small isotropic objects p = 0.90	0.40	0.34	0.52
Large isotropic objects p = 0.10	0.72	0.24	0.72
Large isotropic objects p = 0.50	0.59	0.31	0.63
Large isotropic objects p = 0.90	0.26	0.17	0.63
Small isotropic objects p = 0.10	0.75	0.25	0.66
Small isotropic objects p = 0.50	0.60	0.37	0.59
Small isotropic objects p = 0.90	0.33	0.26	0.58
Large isotropic objects p = 0.10	0.68	0.22	0.74
Large isotropic objects p = 0.50	0.59	0.29	0.65
Large isotropic objects p = 0.90	0.25	0.14	0.64
Real example - One threshold p = 0.50	0.85	0.85	0.35
Real example - Ten thresholds	0.82	0.78	0.29

Table 4.2: Fraction of the nodes updated where $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ was outside the range of $P(\mathbf{A}|\mathbf{B})$ and $P(\mathbf{A}|\mathbf{C})$.

4.7 Quantifying Non-Convexity on the Estimators

One of the good properties of the estimators, when integrating information from various sources is the possibility to be non-convex, although in the kriging context, this is sometimes deemed inappropriate to estimate probabilities. The idea is that the new method performs better than all the individual sources of information when estimating the conditional probability at a location of interest.

Bayesian updating under the full independence and permanence of ratios assumptions gives a unique map of the updated probability given the marginal probability of the event of interest. That is, given $P(\mathbf{A})$, the map of $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ as a function of $P(\mathbf{A}|\mathbf{B})$ and $P(\mathbf{A}|\mathbf{C})$ is fixed (for example, see **Figure 4.22**).

The non-convexity can easily be obtained for these methods, by coding with a different color all the area of this map where the resulting probability $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ is outside the range defined by the two sources of information $P(\mathbf{A}|\mathbf{B})$ and $P(\mathbf{A}|\mathbf{C})$. These maps are presented in **Figure 4.23**.

The non-convexity of all the methods can also be quantified during the simulation procedure. At every node, the three probabilities $P(\mathbf{A}|\mathbf{B})$, $P(\mathbf{A}|\mathbf{C})$, and $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ are known for all the updating methods. The proportion of the time in which the corresponding algorithm generates a result outside the range of the two input probabilities can be used as a measure of non-convexity.

Table 4.2 shows the result for the first two binary examples previously presented. In general, the linear combination of estimates under the multi-Gaussian assumption generated more estimates outside the range of the probabilities inferred from the two sources **B** and **C**.

4.8 Discussion

Updating using different sources of information is difficult because the redundancy between the sources is hard to quantify and therefore some assumption of dependence must be done. The most straightforward approach is to assume that both sources are fully independent, so they provide completely new information. This is not a good assumption in the context of spatial simulation, since in general the

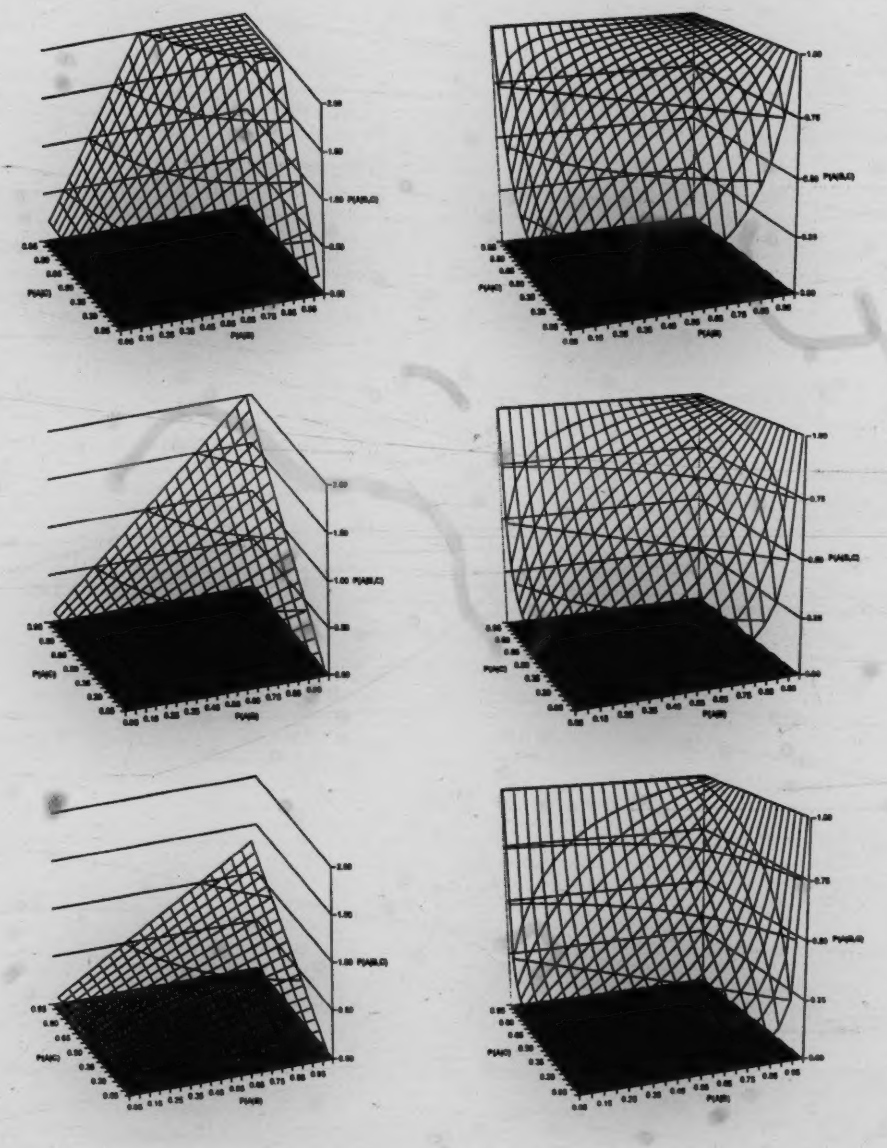


Figure 4.22: Graphs of $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ given $P(\mathbf{A}) = 0.25$ (top), $P(\mathbf{A}) = 0.50$ (middle), and $P(\mathbf{A}) = 0.75$ (bottom). The plots on the left show the maps under the assumption of full independence between **B** and **C**. The plots on the right show the maps under the assumption of permanence of ratios

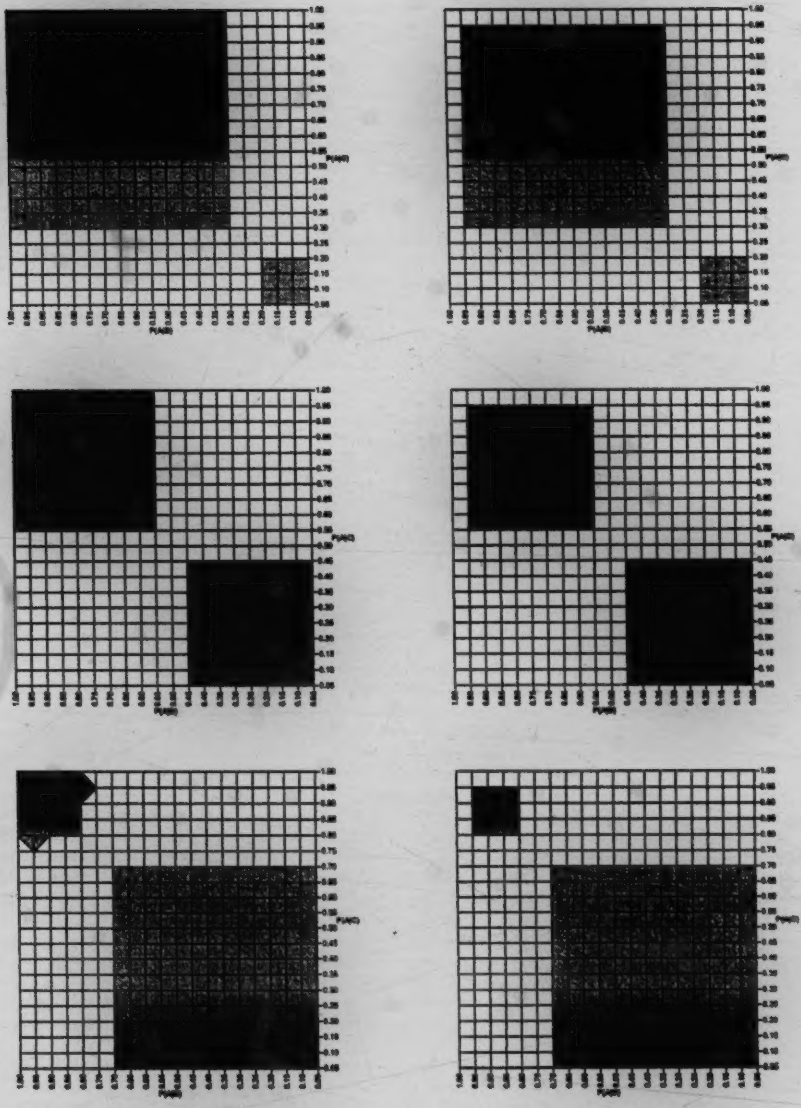


Figure 4.23: Graphs showing in grey the area where the estimated probability $P(\mathbf{A}|\mathbf{B},\mathbf{C})$ is outside the range defined by $P(\mathbf{A}|\mathbf{B})$ and $P(\mathbf{A}|\mathbf{C})$ given $P(\mathbf{A}) = 0.25$ (top), $P(\mathbf{A}) = 0.50$ (middle), and $P(\mathbf{A}) = 0.75$ (bottom). The plots on the left show the maps under the assumption of full independence between **B** and **C**. The plots on the right show the maps under the assumption of permanence of ratios

sources of information are correlated.

The assumption of permanence of ratios has several nice properties that suggest it should perform better than the full independence assumption for updating a conditional probability. The examples presented here showed a similar performance of this updating technique.

Linearly combining the estimates of indicator kriging with the farthest data and the probability of the MP event constituted by the closest (adjacent) nodes to the location being simulated calls for an assumption of dependence. The multi-Gaussian framework is used because of its convenient mathematical properties. The estimation of MP covariances is done by combining two-point covariances. This greatly simplifies the integration of information and allows the calculation of factors (or weights) to be assigned to each estimate. The resulting probability is often outside the range of the input probabilities, which is a nice property, since it means that the updated probability is better informed than the two input probabilities. The examples presented showed a reasonably good performance of this method.

Chapter 5

Case Study

This chapter presents a study undertaken with data from a porphyry copper mine in Chile. The objective is to show the practical implementation of the methods proposed to update the indicator kriging probabilities with multiple-point statistics in a sequential simulation framework and compare the results in terms of medium and long-term planning performance.

The study is introduced in **Section 5.1**. A description of the available data and basic statistics of the drillhole and the blasthole database are presented in **Section 5.2**.

Variogram inference and modelling is presented in **Section 5.3**. The inference of multiple-point statistics from data is illustrated in **Section 5.4**. Classical sequential indicator simulation is performed. The results are shown in **Section 5.5**. The alternative methods for updating the indicator probabilities with multiple-point statistics are implemented in **Sections 5.6**, **5.7**, and **5.8**. Sequential Gaussian simulation is also implemented. The steps are quickly explained in **Section 5.9**. Finally, the results are compared in **Section 5.10**.

5.1 Introduction

The objective of this case study is to show the improvement that can be achieved by considering additional information as multiple-point configurations when generating grade models for mine planning. The methods are compared in terms of mismatch with a kriged model with dense blasthole data, which are used for validation only. Conventional sequential Gaussian simulation is also implemented to compare the proposed method with the most widely applied method in mining.

The classical sequential indicator simulation is compared with the techniques proposed in this research. Updating the indicator kriging probabilities with multiple-point information obtained from production data (blastholes) is done under different assumptions of the relationship between this information and the one that comes from the drillhole samples.

Models are built to match the reference statistics as closely as possible, in expected terms. The same parameters are used for all methods. Corrections due to inconsistency of the two sources of information are implemented to avoid the introduction of bias in the resulting proportions below each threshold.

Blasthole data from two benches are kept aside for validation of the results and comparison between the different methods. Performance is measured by the expected correlation coefficient between the validation data and the simulated grades from each method and by calculation of quantity of metal as compared with a map of the "true" grades, obtained by kriging with dense data.

5.2 Available Data and Basic Statistics

Two databases are available for this study.

A drillhole database is provided that contains over 2300 data. The data are located within the volume defined by the 24450 and 24850 East coordinates, 25000 and 25650 North coordinates, and 3820 and 3950 elevation coordinates (all coordinates in metres). The data are fairly regularly spaced. The approximate drillhole spacing is 50 m. About half of the drillholes are vertical. Plunge and trend are variable for the remaining drillholes.

The second database contains blasthole information for almost 21000 locations. The coordinates range from 24400 to 25000 in the East direction, 24950 to 25800 in the North direction, and 3800 to 4050 in elevation. Blast holes are almost regularly spaced on a squared grid with 10 m separation distance.

5.2.1 Drillhole Information

The drillhole database has composites of length equal to 12 m., which corresponds to the bench height. Drillhole samples typically allow a fundamental sampling error of up to 5%.

This data base contains East, North and elevation coordinates, the copper grade in percent by weight, and the rock type code. Seven different rock type codes exist: 4, 20, 28, 29, 31, 34, and 54. However, the only rock type of interest is 20, since this is the code of the material of economic interest. The study will be done considering only these data. Furthermore, it was found that data over elevation 3928 has a larger local variance (see **Section 5.2.4**). All the data within rock type 20 below this elevation can be considered belonging to an homogeneous population, where quasi stationarity of the mean and variance appears as a reasonable assumption. For this reason, the study considered only data below elevation $Z = 3928$ m.

Figure 5.1 shows the histograms of composites including all rock types and elevations, and considering only the composites with rock type coded as 20 under elevation 3928. 2376 composites in total and 1281 in rock type 20 below elevation 3928 are available. The average grade within the reduced domain is higher than when considering all data. The data range from 0 to around 7 %Cu and the distribution is positively skewed. The coefficient of variation is approximately 0.5, which can be considered relatively low. It is a normal value for deposits of this type. The median is very close to the mean value.

Probability plots are presented to compare the distribution of grades with a lognormal distribution (**Figure 5.2**). A very good fit of a straight line could be done, except for the upper part of the curve, where there is a shift on the slope. High grades have a different behavior than low grades. The curve does not change much by considering the reduced domain (rock type 20 and elevation below $Z = 3928$). The same shift in the slope is seen in this plot.

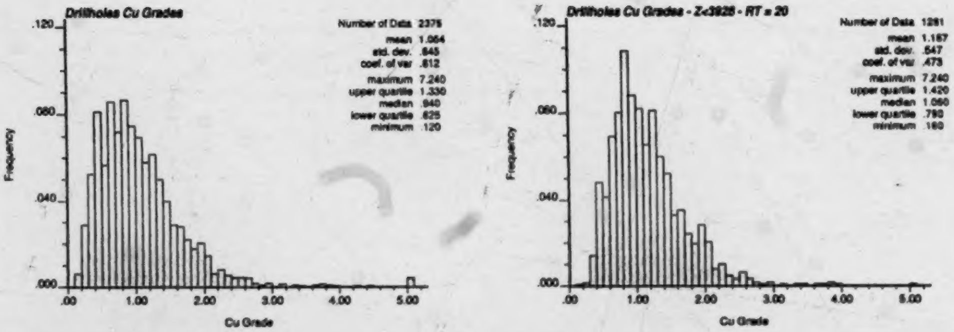


Figure 5.1: Histogram of copper grade considering all composites (left) and only composites with rock type code 20 and under elevation 3928 (right).

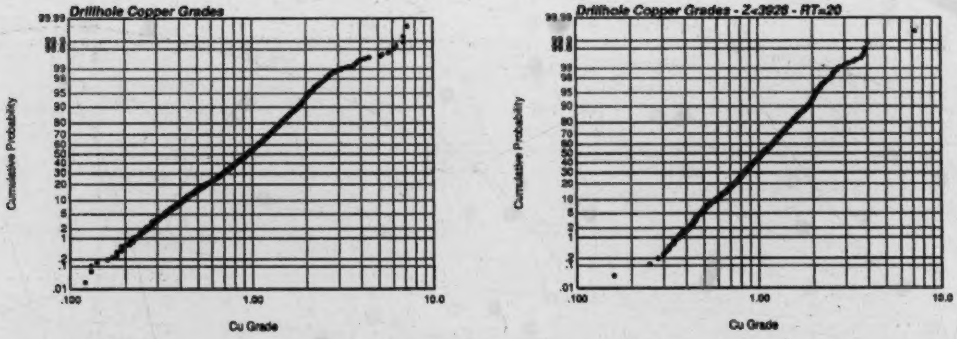


Figure 5.2: Probability plots for the entire copper grade dataset (left) and for the samples in rock type 20 and under elevation 3928 (right). The distribution appears close to a lognormal, with a slight change in the high values.

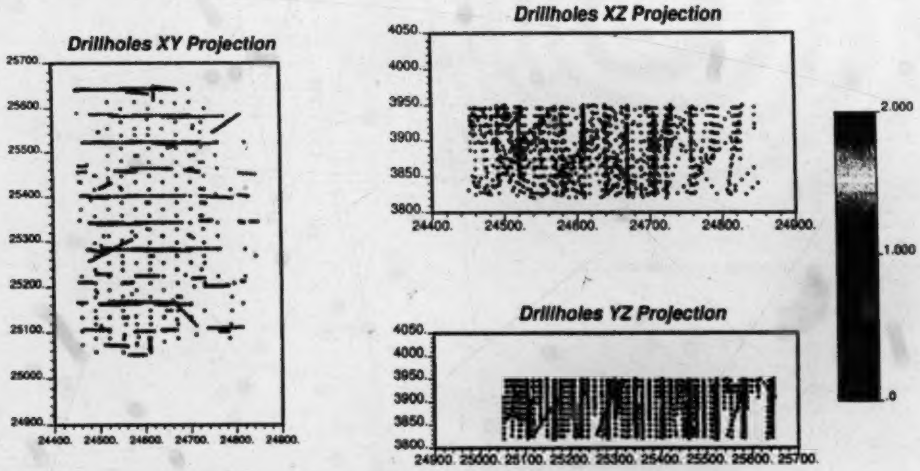


Figure 5.3: Projection over the three planes horizontal, vertical along the East-West direction, and vertical along the North-South direction, showing the drillhole data.

Drillholes are shown in **Figure 5.3**. These maps are projections of all the samples on the horizontal plane, vertical plane along the East-West direction, and vertical plane along the North-South direction. Different directions for the drillholes can clearly be seen on these projections. Notice that in the horizontal projection, the vertical drillholes appear as a point, since the projections of all the samples of the drillhole fall in the same point on the plane.

Some plan views are presented in **Figure 5.4**. The copper grades are shown at the bench level with a tolerance of 12 m. Many drillholes have been drilled in this mine because it is currently in production. The average spacing between drillholes is around 50 m. In many zones drillholes are spaced even closer.

Rock type codes are shown for these same plan views on **Figure 5.5**. The samples coded with 20 are shown in white, while the other rock types are shown in black. It is clear that the other codes are located mainly in the boundaries of the mineralization.

5.2.2 Blasthole Information

Blastholes for several benches are available. Blastholes are typically drilled at the bench height plus about 10% of sub-drill depth. However, the samples are taken once the bench height (12 m) is reached, that is, the sub-drilling is not included in the sample. Diameter is typically 9 3/4 inches. Sample protocols for blasthole material generate a fundamental Cu grade error of up to 15%. Although the support is larger than the one of the drillhole samples, this larger sampling error increases the variance of these data.

A histogram of the blasthole data considered for this study (benches 3886 to 3922) along with a lognormal probability plot are presented in **Figure 5.6**. Plan views of some of the benches are presented in **Figure 5.7**. The samples appear quite

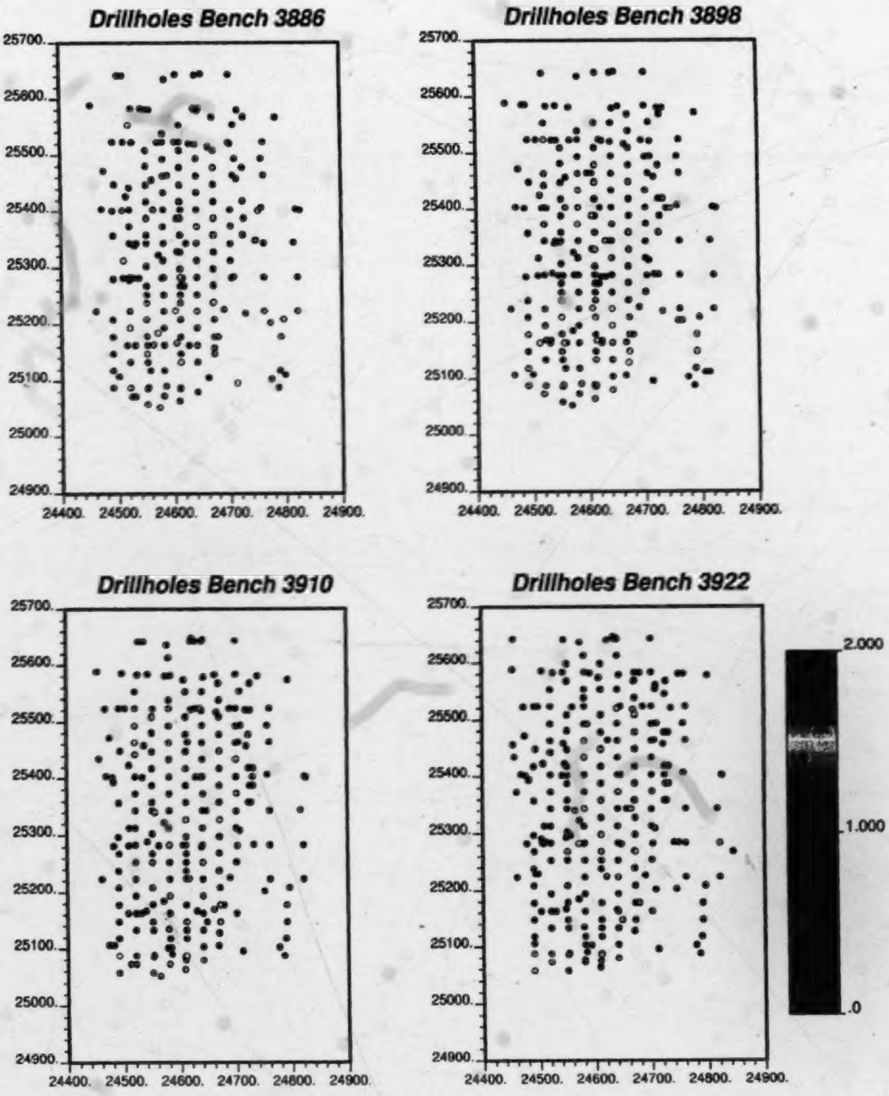


Figure 5.4: Plan views showing the drillhole information.

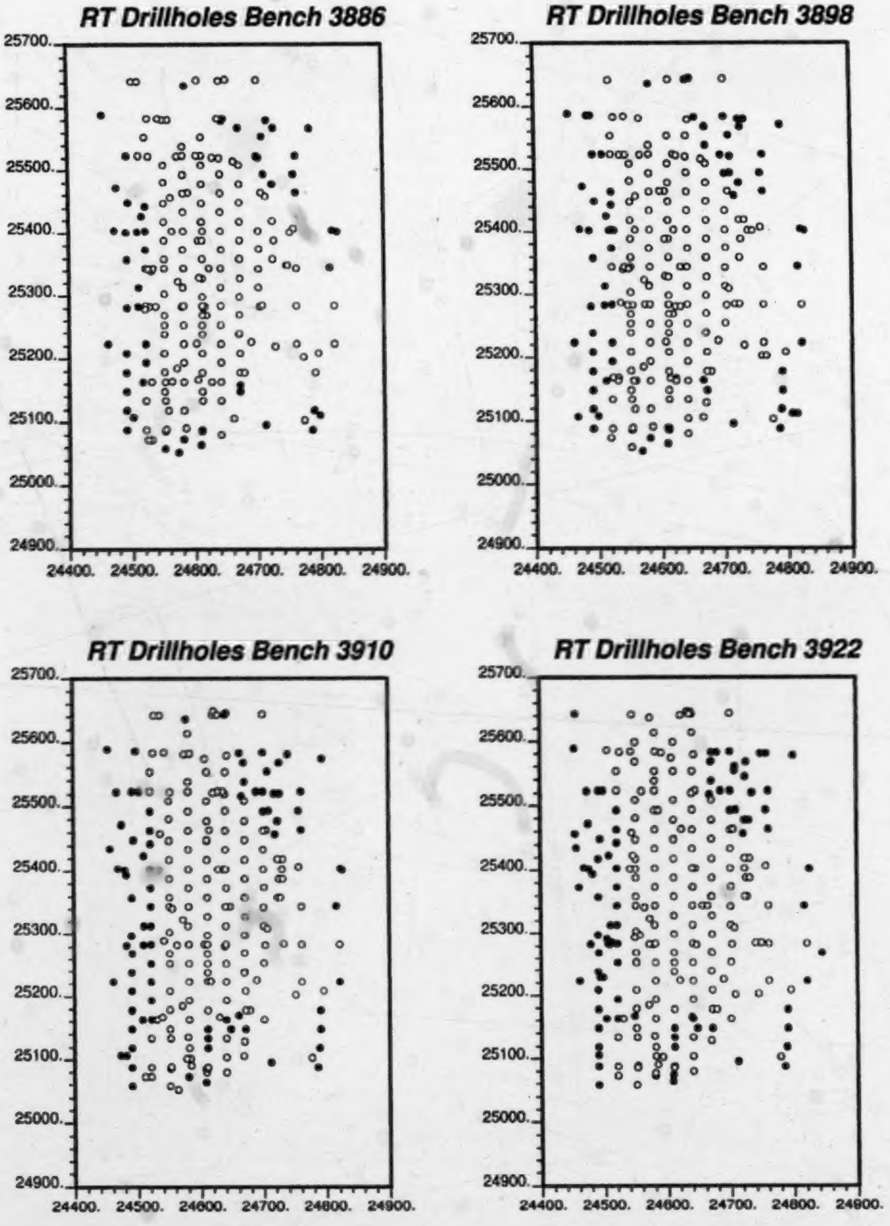


Figure 5.5: Plan views showing in white the locations of drillhole samples with rock type code 20 and, in black, samples with other codes.

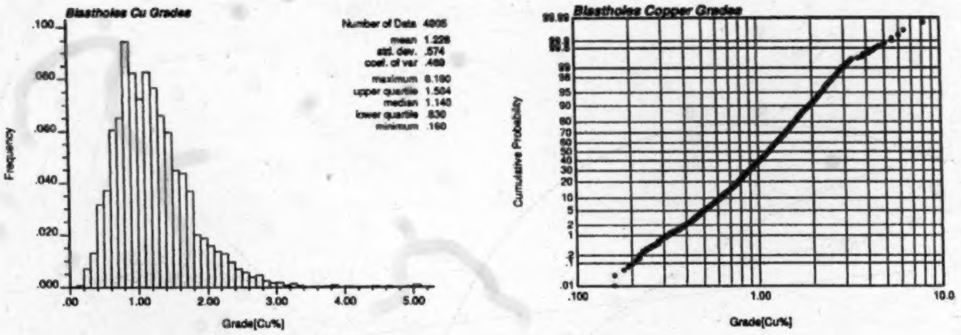


Figure 5.6: Histogram and lognormal probability plot of copper grade from the blastholes.

exhaustive and regularly spaced. Blastholes are more irregular in the perimeter where damage control on the walls requires a closer spacing and delineating the wall.

For comparison, the two lower benches (elevations 3886 and 3898) will be kept aside of all inference and estimation procedures, since they will be used for the final comparison of performance of the methods.

Statistics for the remaining data are presented in **Figure 5.8**.

5.2.3 Declustering

Declustering is required to obtain a representative reference distribution for simulation. Although the drillhole samples appear quite regularly distributed in space, the different orientations of the drillholes and the fact that they are distributed in the three dimensional space, it is hard to judge visually if high grade or low grade zones have been over sampled. A cell declustering procedure is applied to find the cell size that minimizes the mean. Given the spacing of the data an anisotropic cell is used with a ratio horizontal to vertical size of 4 to 1, since the vertical spacing of the samples is 12 m and the drillhole spacing is approximately 50 m. The most appropriate cell size to obtain a representative distribution was $120 \times 120 \times 30\ m^3$ (**Figure 5.9**). The representative histogram obtained by considering the samples with the declustering weights is shown in **Figure 5.10**. Notice the reduction on the mean value from 1.157 %Cu to 1.068 %Cu, and that the variance remained almost constant. The change in the declustered mean is within the normal range seen in this type of deposit. The reduction in the mean is less than 10% of the average clustered grade, meaning that clusters had not a large impact in the global statistics.

The declustering weights will be used to correct the cumulative distribution function value below each threshold.

5.2.4 Comparison of Datasets

The drillhole and blasthole information can be compared in several ways:

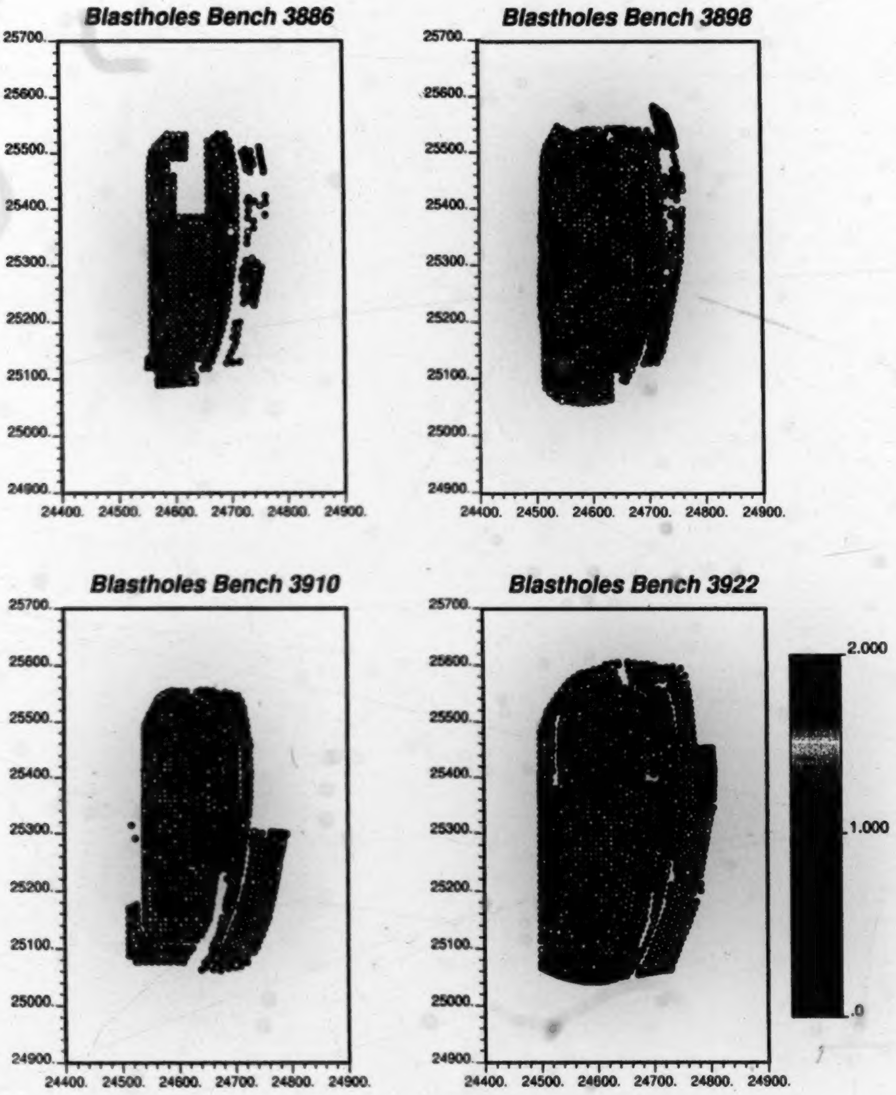


Figure 5.7: Plan views showing the blasthole information.

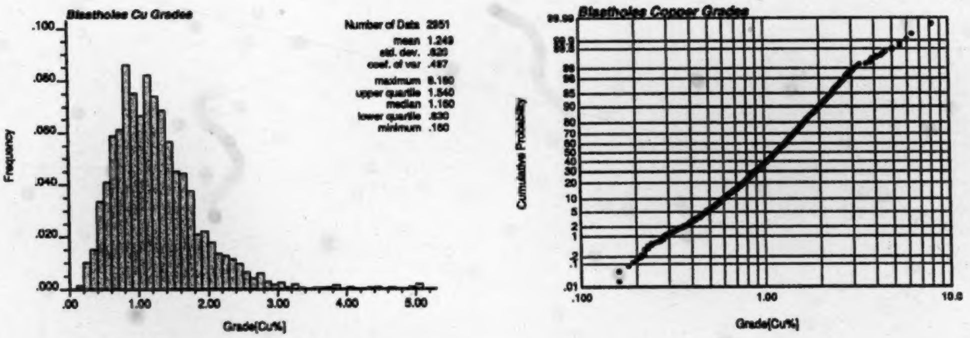


Figure 5.8: Histogram and lognormal probability plot of copper grade from the blastholes of benches 3910 and 3922.

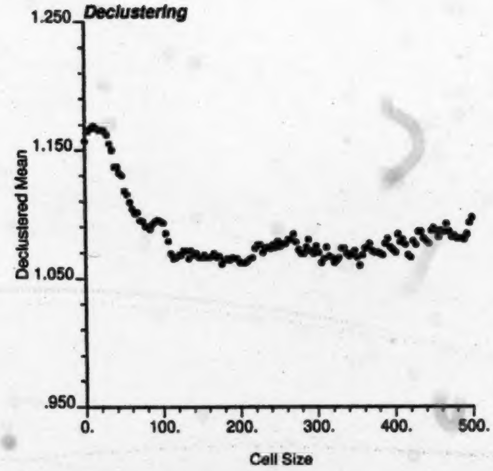


Figure 5.9: Cell size versus declustered mean. An optimum cell size of 120 m in horizontal directions is considered.

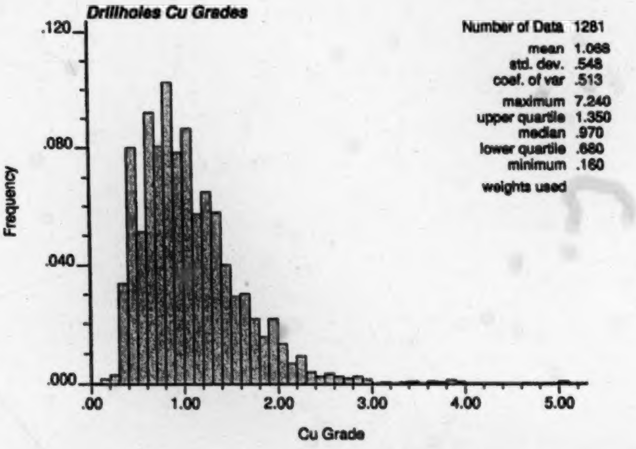


Figure 5.10: Histogram of declustered copper grade from the drillhole data with rock type code 20 and elevations below 3928.

Global distribution The two global distributions can be compared with a q-q plot. A q-q plot is a cross plot of pairs, where the first element of the pair is the quantile value of one distribution and the second element, the same quantile from the second distribution. For example, one pair will be formed by the value at which 1% of the data falls under that value for each distribution. If the distributions are equal, the plot will look as a straight line at 45 degrees and intersecting the origin. Any departure from the 45 degrees reflects differences in the distributions.

Spatial distribution Each drillhole data can be associated to the closest blasthole samples, within some maximum distance. A cross plot of the pairs will give an idea of the match of the two datasets in a spatial context. The correlation coefficient should be high, although not one, since there are sampling errors for both types of samples, and there is a distance tolerance which will make the correlation decrease.

Trends In average terms, drillholes and blastholes should show the same trends when looking at moving averages on different directions.

The q-q plot is shown in **Figure 5.11**. Notice that both distributions match very closely up to a grade of approximately 3.0 %Cu. Although over this value the discrepancy is larger, this corresponds to less than 1 % of the population (see **Figures 5.2** and **5.6**). Notice that the domain over which drillholes are distributed is much larger than the volume informed by blastholes.

Drillhole and blasthole data were paired considering different tolerance distances, that is, for a given drillhole sample, all blasthole samples falling within that distance were identified and subsequently, a cross plot of the pairs was plotted. **Figure 5.12** shows the cross plots for three increasing tolerances. As expected, the number of pairs found increases as the tolerance distance is increased. Also, the correlation coefficient decreases with larger tolerances, since the samples tend to be less corre-

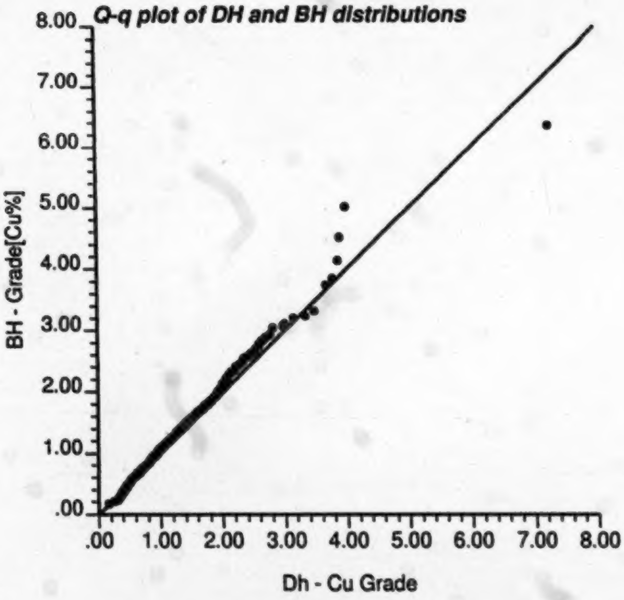


Figure 5.11: Q-Q plot of drillhole copper values and blasthole sample values.

lated the farther away they are. Notice that, due to sampling errors and the nugget effect on the copper grade, these cross plots must show a correlation lower than 1.0 even in the case the samples are very close. Thus, a correlation coefficient of 0.7 reflects a very good match between the two data sets. Also notice that the mean of drillhole and blasthole data tend to get closer as more samples are used. When a 10 m tolerance is used to pair the data, the means are almost equal.

Moving averages are calculated over stripes along the three main directions (easting, northing, and elevation) to check for abrupt changes in the local mean and variance. The general trend should look the same for both sets of data. This analysis has two purposes. First, it allows comparing the two datasets. Second, it can be used to assess stationarity of the data for the subsequent geostatistical simulation method. In case of finding sudden changes over short distances in the local mean and variance, the trend should be removed to work with the stationary residuals. Plots of the local mean and variance along the three main directions are presented in **Figures 5.13**, **5.14**, and **5.15**. The results presented in these trend graphs show that the two sets of data behave similarly regarding changes on the local means and variances, except for elevations over $Z = 3928$. This is the reason to discard the data above this elevation for statistical inference.

5.2.5 Comments

From the previous analysis, it seems reasonable to utilize the blasthole data from the benches 3886 and 3898 for comparison of the results.

The drillhole data above the elevation $Z = 3928$ is not considered for statistical inference, since the two data sets give different results. The variations in local means and variances are considered reasonably stationary when considering local

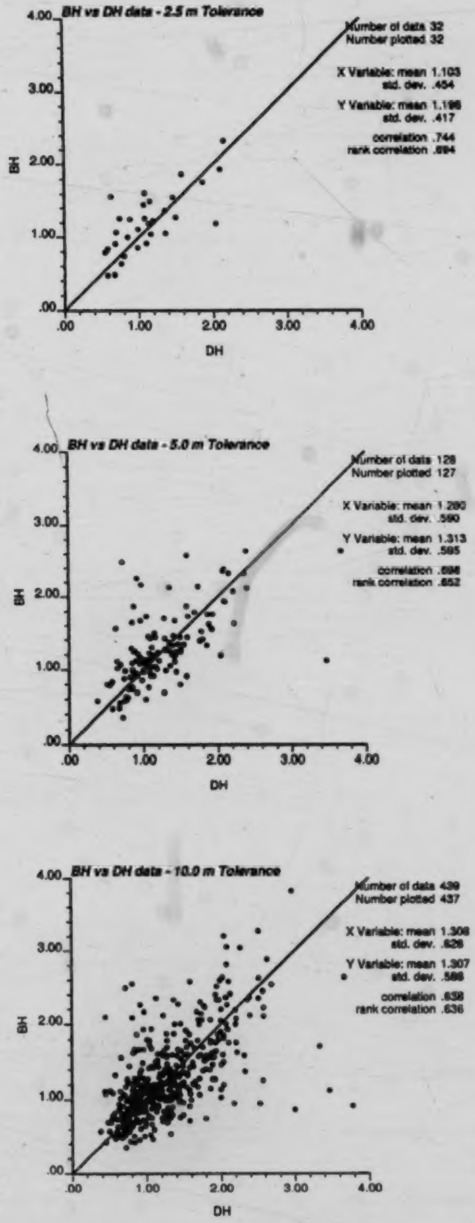


Figure 5.12: Cross plots of paired samples for different tolerance distances.

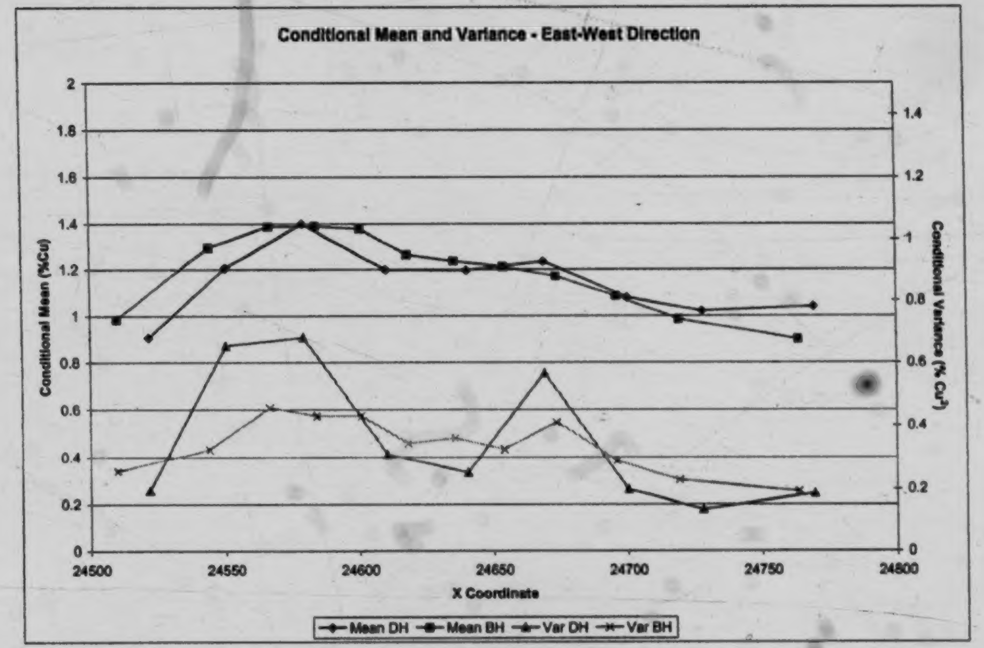


Figure 5.13: Local mean and variance along the East-West direction of the drillhole and blasthole data.

neighborhoods.

Although relatively large differences can be seen in the global means of drillhole and blasthole sample grades, these can be rationalized by considering the larger volume informed by the drillhole data compared to the volume informed by blastoles. Blastholes tend to inform only the center of the deposit, which has a higher grade than the boundaries, better informed by drillholes. Furthermore, the datasets have been validated at the mine and there is no evidence of a systematic bias in the blasthole data. Discrepancies will be dealt with when updating to avoid a bias due to the difference in the means (see **Section 5.7**).

Differences in the high grades are deemed minor and will not be reflected in the indicator simulation, except during the step of extrapolating beyond the upper tail. Care must be taken to avoid a bias due to extrapolation of the upper tail. The distribution of copper values in the drillhole data set will be used as the representative distribution, considering the weights from declustering to correct the proportions below each threshold, since these samples have a lower sampling error than blasthole samples.

There is no need to model and remove the trend, since local stationarity appears as a reasonable assumption. Sequential indicator simulation should perform well under these circumstances.

The calculation of indicator variograms follows and then inference of multiple-point statistics from the blasthole data set, without including the data belonging to the benches that are used for validating the final results.

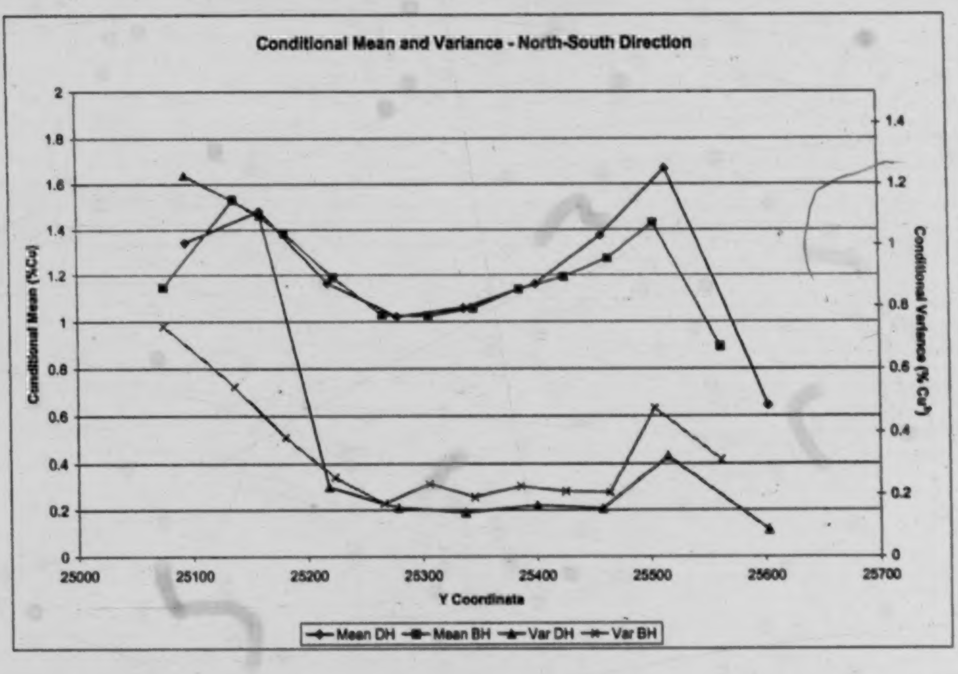


Figure 5.14: Local mean and variance along the North-South direction of the drillhole and blasthole data.

5.3 Variogram Modelling

5.3.1 Selection of Thresholds

To obtain an adequate discretization of the conditional distributions, 10 thresholds are used in all subsequent calculations involving indicators. The selection of these 10 values calls for several considerations: the full distribution should be adequately sampled by these values, that is, selecting values that are regularly spaced (in terms of probabilities) is convenient because interpolation between thresholds does not carry many difficulties; the adequate characterization of high grades is required, hence additional thresholds are located in the high tail of the distribution, however, inference becomes more difficult as the threshold is more extreme. The 10 threshold values correspond to the nine deciles in the clustered distribution, and an additional threshold at the quantile 0.95. This last value will help characterizing the high values, minimizing extrapolation problems due to the skewness of the distribution.

The proportions below the thresholds considering the declustering weights are used within the indicator simulation.

Table 5.1 shows the threshold values, proportions that fall below that threshold in the clustered distribution, and the proportions corrected to account for the clusters.

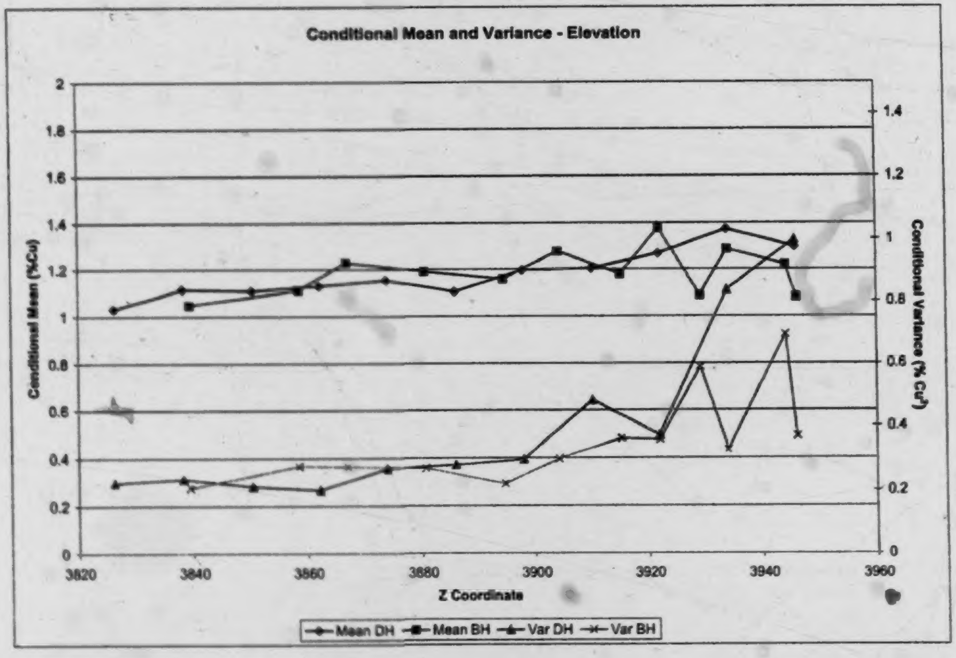


Figure 5.15: Local mean and variance with elevation of the drillhole and blasthole data.

Threshold number	1	2	3	4	5	6	7	8	9	10
Threshold value	0.58	0.73	0.84	0.95	1.08	1.22	1.36	1.56	1.91	2.18
Clustered quantile	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95
Declustered quantile	0.15	0.28	0.38	0.47	0.57	0.68	0.76	0.85	0.93	0.97

Table 5.1: Threshold definition for indicator variogram calculation and simulation

Parameter	Horizontal Directions	Vertical Direction
Number of lags	10	5
Lag Separation Distance	15.0	12.0
Lag Distance Tolerance	7.5	6.0
Azimuth Tolerance	22.5	22.5
Horizontal Bandwidth	25.0	25.0
Dip tolerance	22.5	22.5
Vertical Bandwidth	25.0	25.0

Table 5.2: Parameters for calculation of experimental variograms

5.3.2 Variogram Calculation and Modelling

Anisotropy

To determine the main directions of anisotropy, variograms were calculated in several directions (not shown). The main directions of anisotropy were found at N30°W, N60°E, and vertical. This information is consistent with geological information of the region.

Experimental Variogram Calculation

Variograms were calculated in the three principal directions: N30°W, N60°E, and vertical. The other parameters used to calculate the experimental indicator variograms are shown on **Table 5.2**. The data used for variogram inference correspond to the drillhole information for rock type 20 and below the elevation 3928, and the blasthole samples taken in the benches 3910 and 3922. Given the abundant information, variograms could be calculated with relatively small tolerances. This should ensure that they correspond to the directions of interest and that averaging with other directions by increasing the tolerances is avoided.

Variogram Modelling

Variogram modelling is done considering that abrupt changes in the model from one threshold to the adjacent will generate order relation deviations, which are undesirable. Therefore, the modelling process takes into account the adjacent variogram models, so that any change is consistent from one threshold to the next.

Table 5.3 shows the parameters for the models fitted to the experimental variograms. The fitting is presented in **Figure 5.16**. Three structures are used to model the variogram: two spherical and one exponential. The nugget effect is smaller for thresholds far from the median, opposite to what is obtained using a multi-Gaussian method. Inference of the variogram at the lowest threshold gave an erratic experimental variogram, particularly for short distances. The nugget effect and ranges were considered based on the variogram at the next threshold. Ranges tend to decrease as the cutoff increases, which is common in metal concentrations such as gold, silver, and in a lesser extent, copper.

Changes in sill can be seen in **Figure 5.17**. Changes in ranges of the variograms are presented in **Figure 5.18**.

		Spherical				Spherical				Exponential			
Cutoff	Nugget Effect	Sill	Range			Sill	Range			Sill	Range		
			N30°W	N60°E	Vert.		N30°W	N60°E	Vert.		N30°W	N60°E	Vert.
0.58	0.30	0.25	25.0	40.0	30.0	0.27	480.0	380.0	45.0	0.18	∞	280.0	∞
0.73	0.30	0.25	25.0	40.0	20.0	0.27	200.0	220.0	30.0	0.18	∞	200.0	∞
0.84	0.30	0.25	25.0	40.0	25.0	0.25	200.0	140.0	35.0	0.20	320.0	180.0	∞
0.95	0.30	0.25	40.0	70.0	30.0	0.25	160.0	100.0	40.0	0.20	180.0	120.0	∞
1.08	0.35	0.20	40.0	65.0	40.0	0.25	130.0	85.0	130.0	0.20	130.0	80.0	130.0
1.22	0.35	0.20	40.0	35.0	50.0	0.25	90.0	85.0	130.0	0.20	110.0	80.0	130.0
1.36	0.30	0.25	35.0	30.0	60.0	0.25	80.0	65.0	130.0	0.20	90.0	75.0	130.0
1.56	0.30	0.30	35.0	30.0	60.0	0.20	70.0	65.0	140.0	0.20	60.0	55.0	140.0
1.91	0.25	0.40	25.0	20.0	50.0	0.15	60.0	55.0	150.0	0.20	60.0	55.0	150.0
2.18	0.20	0.40	20.0	20.0	28.0	0.15	35.0	35.0	∞	0.25	40.0	40.0	∞

Table 5.3: Indicator variogram model parameters.

Direction	Number of nodes	Center coordinate of first node	Grid spacing
Easting	50	24405.0	10.0
Northing	80	24905.0	10.0
Elevation	2	3910.0	12.0

Table 5.4: Grid definition for multiple-point inference and simulation.

5.4 Multiple-Point Statistics Inference

Blasthole data from benches 3910 and 3922 are used to infer multiple-point statistics. The scattered blasthole locations are associated with the closest point on regular grid from which the frequencies of MP configurations for all the patterns shown in **Figure 4.2**, are inferred.

Inference is made by simply counting how many times there is a one at the central node of the MP configuration, given the indicator values of the four adjacent nodes, if informed. This count is divided by the total number of MP events with the same configuration to approximate the frequency of this event.

Figure 5.19 shows the indicator maps from the blasthole dataset for one bench considering a regular grid defined by the parameters in **Table 5.4**.

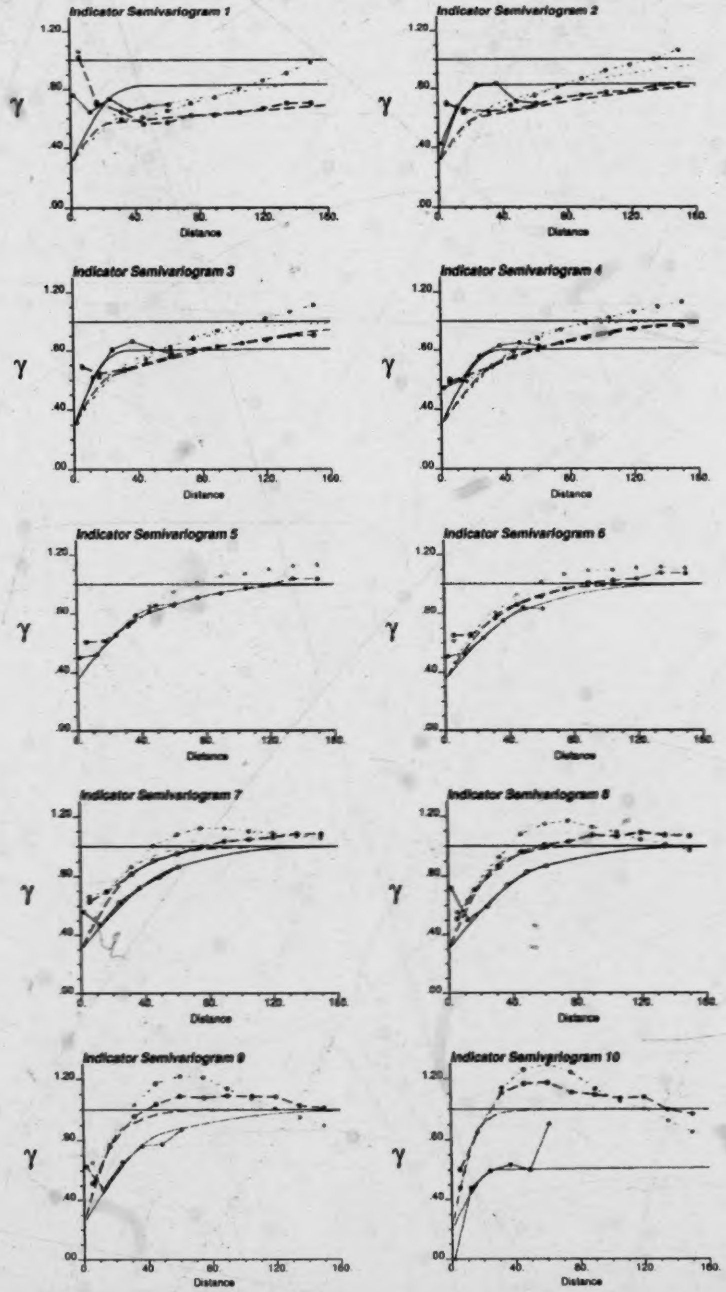
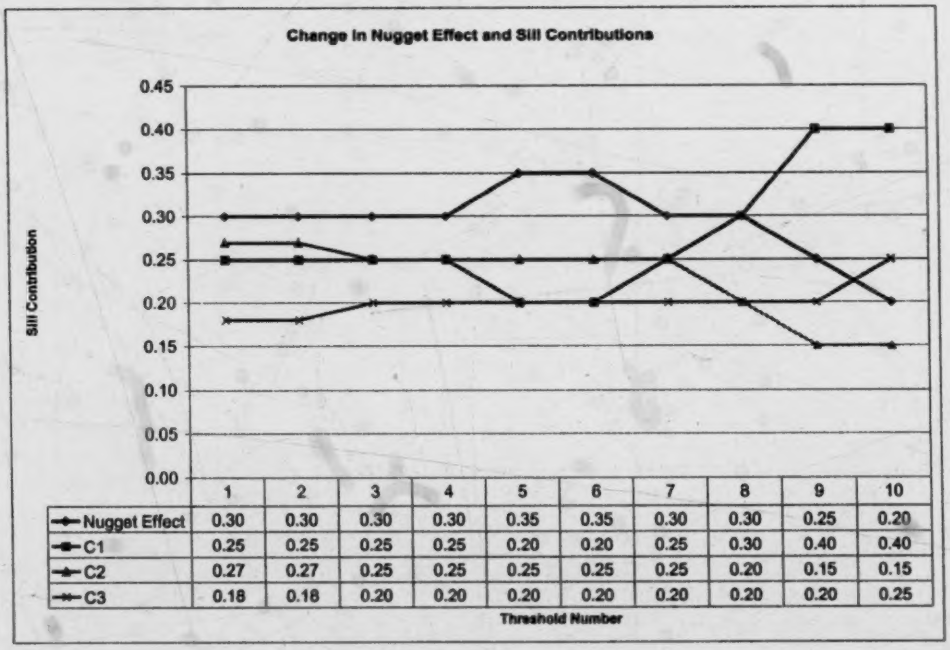
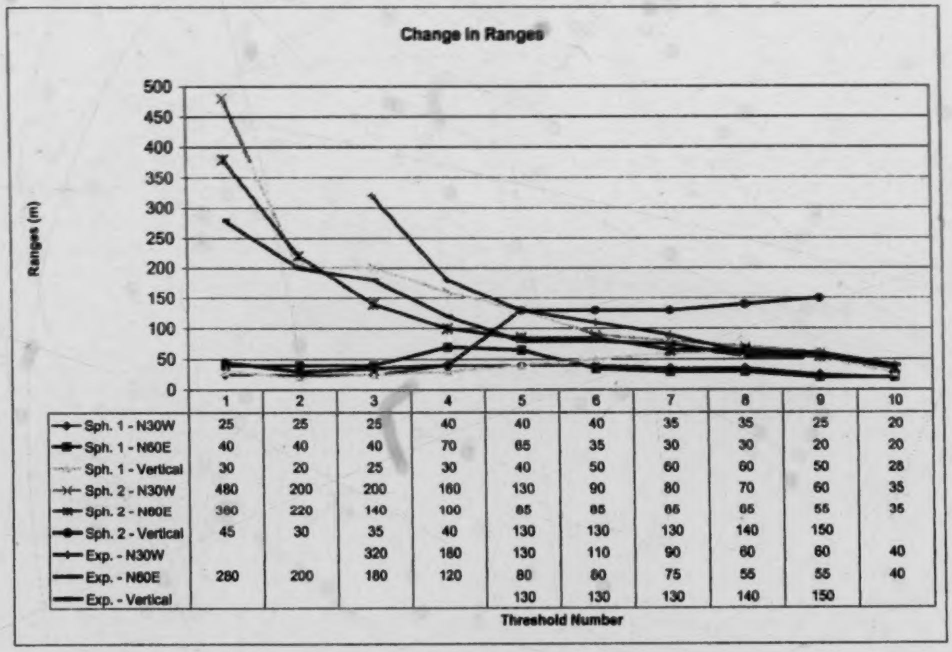


Figure 5.16: Indicator variogram models fitted to the experimental variograms in the three principal directions of anisotropy. The continuous line corresponds to the vertical direction, the dashed line is in the N30°W direction, and the dotted line corresponds to the N60°E direction.



	1	2	3	4	5	6	7	8	9	10
Nugget Effect	0.30	0.30	0.30	0.30	0.35	0.35	0.30	0.30	0.25	0.20
C1	0.25	0.25	0.25	0.25	0.20	0.20	0.25	0.30	0.40	0.40
C2	0.27	0.27	0.25	0.25	0.25	0.25	0.25	0.20	0.15	0.15
C3	0.18	0.18	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.25

Figure 5.17: Change in nugget effect and sill contributions for different thresholds.



	1	2	3	4	5	6	7	8	9	10
Sph. 1 - N30W	25	25	25	40	40	40	35	35	25	20
Sph. 1 - N60E	40	40	40	70	65	35	30	30	20	20
Sph. 1 - Vertical	30	20	25	30	40	50	60	60	50	28
Sph. 2 - N30W	480	200	200	160	130	90	80	70	60	35
Sph. 2 - N60E	380	220	140	100	85	85	65	65	55	35
Sph. 2 - Vertical	45	30	35	40	130	130	130	140	150	
Exp. - N30W			320	180	130	110	90	60	60	40
Exp. - N60E	280	200	180	120	80	80	75	55	55	40
Exp. - Vertical					130	130	130	140	150	

Figure 5.18: Change in ranges for different thresholds.

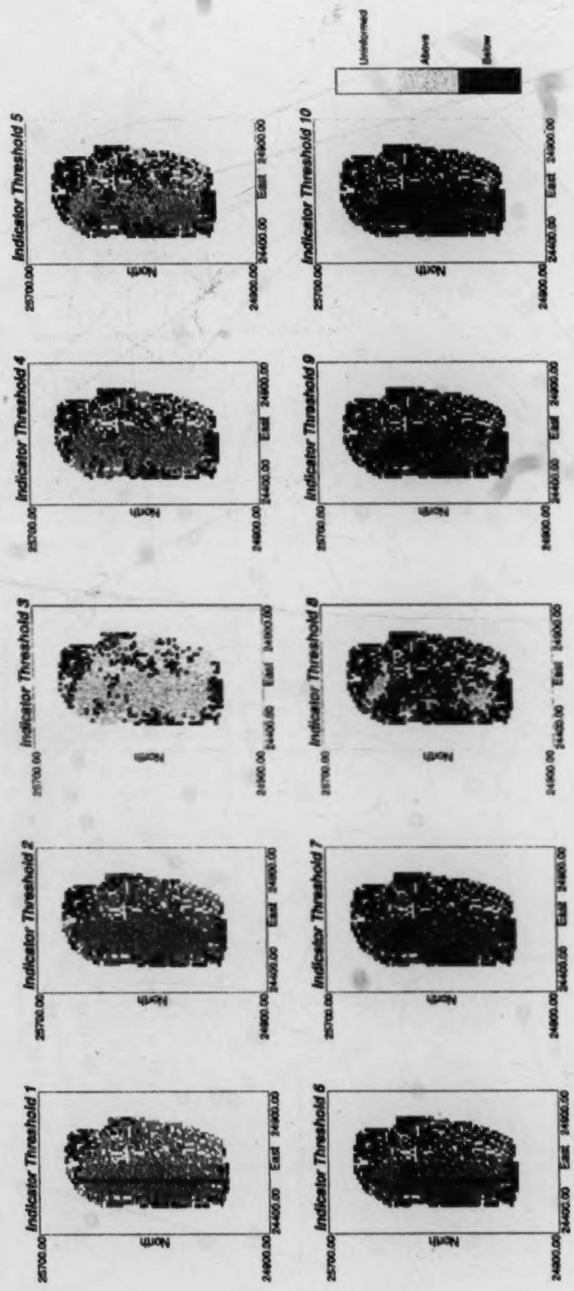

Indicator Threshold 1
Indicator Threshold 2
Indicator Threshold 3
Indicator Threshold 4
Indicator Threshold 5
Indicator Threshold 6
Indicator Threshold 7
Indicator Threshold 8
Indicator Threshold 9
Indicator Threshold 10
North
East
25700.00
24900.00
24400.00
Uninformed
Above
Below

Figure 5.19: Indicator values of the scattered blasthole data approximated by a regular grid.

Random number generator seed	120574
Max. data for kriging	24
Max. previously sim. nodes	24
Multiple-grid search levels	3
Maximum search radius horiz.	300.0 m
Maximum search radius vertical	150.0 m

Table 5.5: Simulation parameters.

Simulation must be done at the same resolution defined on **Table 5.4** if MP information is used to update the indicator kriging probabilities.

5.5 Sequential Indicator Simulation

5.5.1 Parameters

100 realizations obtained by sequential indicator simulation (SIS) are generated (see **Figure 5.20**). Thresholds and corrected proportions presented in **Table 5.1** are used. The conditioning data corresponds to the drillhole samples with rock type 20 under the elevation 3928. Interpolation between thresholds is done linearly, while for the tails, the shape of the global declustered distribution is re-scaled for extrapolation, considering a minimum copper grade of 0.0 % and a maximum of 7.5 %. The grid specification is as defined in **Table 5.4**, but instead of considering the two benches 3910 and 3922, two benches are simulated below these, that is, with elevations 3886 and 3898. The seed for the random number generator and search parameters are presented in **Table 5.5**.

Maps of the two benches for the first two renditions obtained by indicator simulation are presented in **Figure 5.20**.

5.5.2 Validation of Results

Reproduction of Statistics

A histogram and q-q plot of all the realizations considered together are built to check overall performance (**Figure 5.21**). The reproduction of the mean, variance, and quantiles of the reference distribution is acceptable.

The mean and the variance of each realization is calculated and plotted on histograms. The reference values are signaled as black dots underneath the histograms (**Figure 5.22**). This graph shows the good reproduction of the histogram.

Additionally, q-q plots were built for each realization. Some of them are shown on **Figure 5.23**. Quantiles differ for grades greater than 3.0 %Cu, which represent a very small proportion of the population.

Reproduction of Data Values

The data were assigned to grid nodes. The original values were then coded as indicators for all thresholds. The algorithm ensures reproduction of the data values by drawing a simulated value only if the node is not informed, that is, if an original value existed for the node, the algorithm will keep that value rather than drawing a

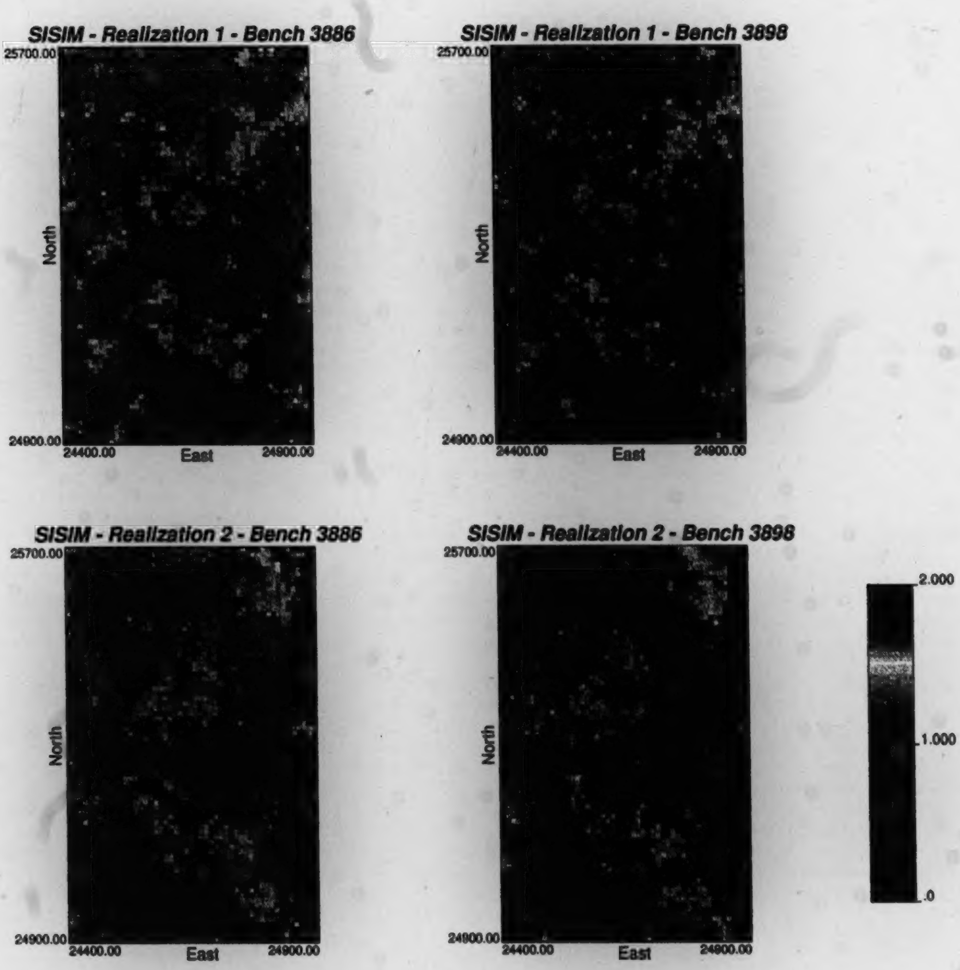

SISIM - Realization 1 - Bench 3886
SISIM - Realization 1 - Bench 3898
SISIM - Realization 2 - Bench 3886
SISIM - Realization 2 - Bench 3898
25700.00
24900.00
North
24400.00
East
24900.00
2.000
1.000
.0

Figure 5.20: Maps of the two benches for the first two realizations by SIS.

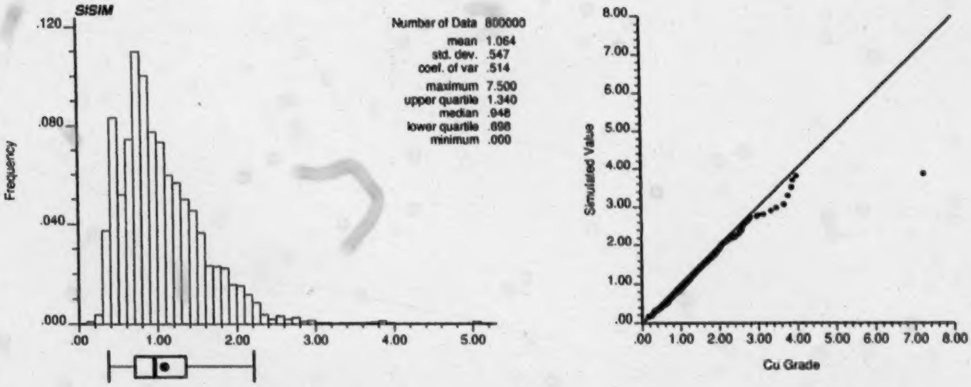


Figure 5.21: Histogram and q-q plot of all the simulated values by SIS (100 realizations). The dot represents the mean from the reference declustered distribution.

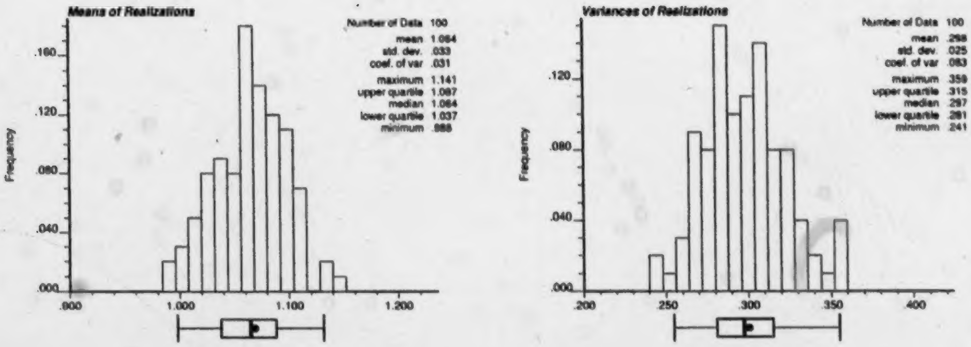


Figure 5.22: Histograms of the means and variances of the realizations by SIS. The dots below the histogram represent the corresponding reference values.

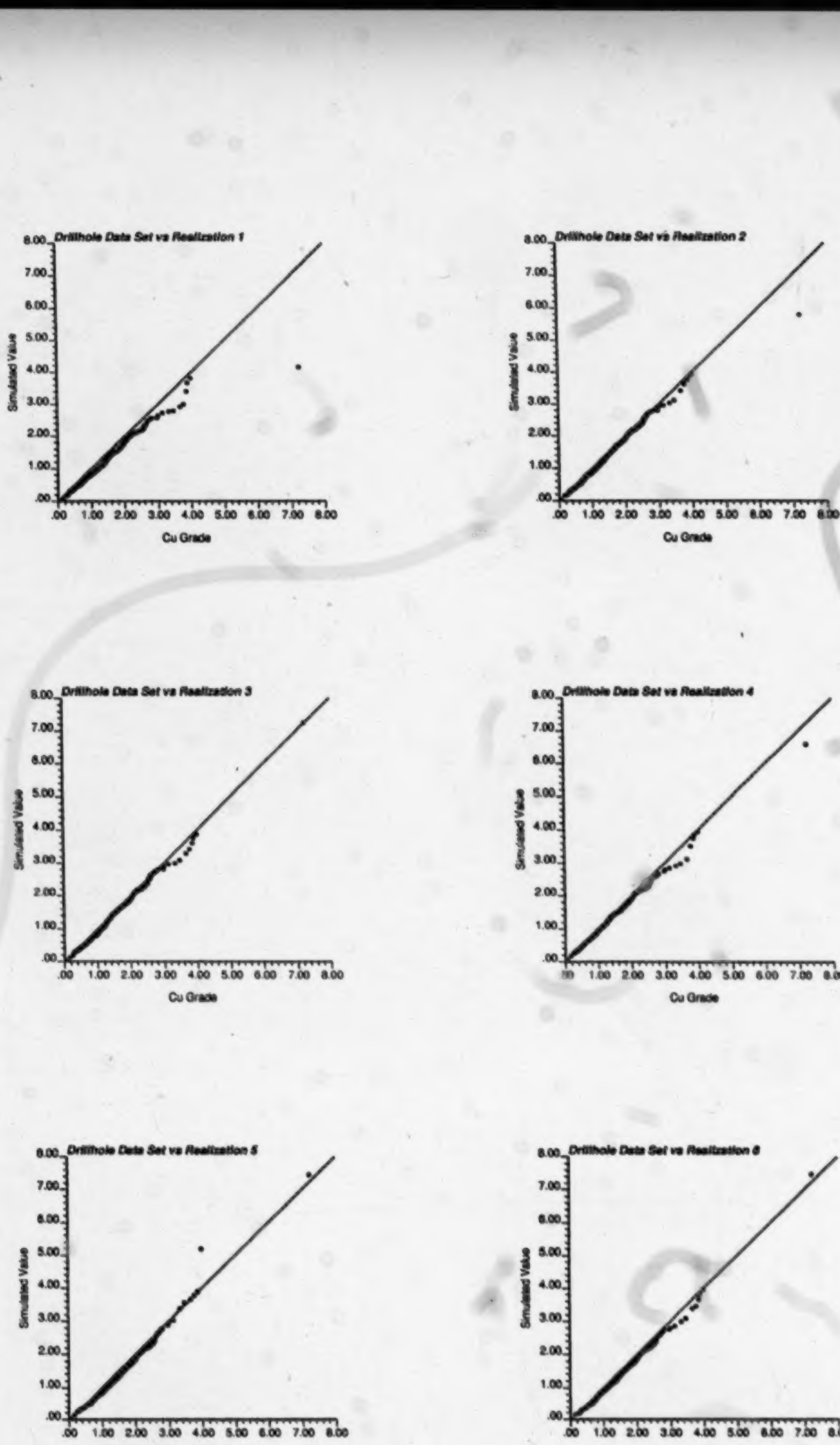


Figure 5.23: Q-Q plots of the reference distribution versus the distribution from the first six simulated models by SIS.

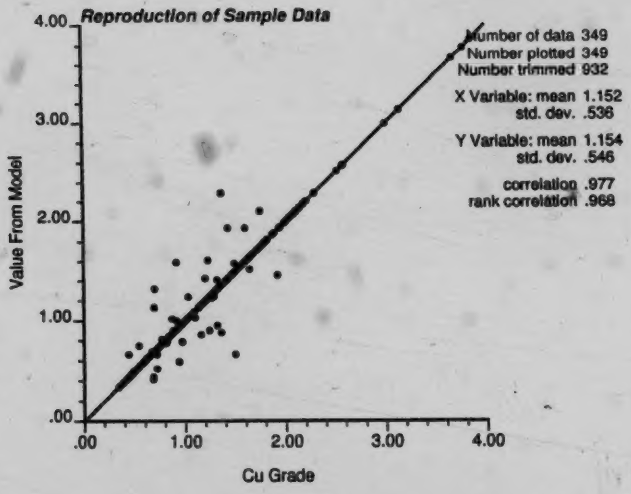


Figure 5.24: Cross plot of sample values and the value assigned at the closest node in the models simulated by `SIS`. Only 307 out of 349 data inside the model are reproduced, since a closer sample is assigned to the node.

new value in the corresponding class defined by the indicators. Although indicator methods have a resolution based on the number of the thresholds defined to code the data, the implementation used ensures that the lost resolution will not affect sample values, hence they are assigned to nodes without drawing a new value.

It is expected that all data will be honored unless more than one sample is assigned to the same node, in which case the closest sample will be assigned. **Figure 5.24** shows the reproduction of sample data. From the 1281 samples available, only 349 are located inside the model defined by the two benches to be simulated, that is, their elevation is between 3880 and 3904. From these samples, 42 are assigned to nodes that have another closest sample, hence their values are not reproduced.

Reproduction of Indicator Variograms

From the models generated, indicator variograms can be calculated approximately for the main directions of anisotropy, due to the grid specification. The vertical direction cannot be checked with the two benches simulated.

Variograms are calculated for lags multiple of a vector defined by one node in the West direction and two nodes in the North direction, which corresponds to an azimuth of 26.5^o. Similarly, the perpendicular horizontal direction is calculated for vectors multiple of a vector defined by two nodes East and one node North, corresponding to 63.5^o (**Figure 5.25**).

The experimental variograms for the two horizontal directions for each realization along with the corresponding model are presented in **Figures 5.26** and **5.27**.

Variograms are well reproduced, except for the first few thresholds in the first direction ($N26.5^oW$), where the range is shorter in the realizations than in the model. However, the shift can be considered minor in all the cases.

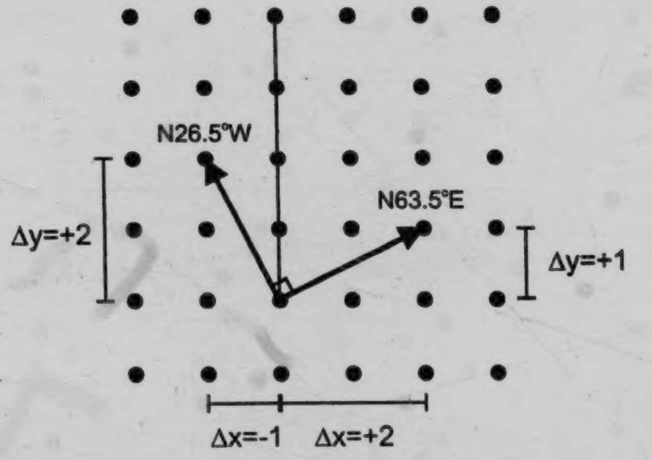


Figure 5.25: Definition of the directions for variogram calculation in the regular grid of the model.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	3436	0.0086	0.1544
2	4256	0.0138	0.1986
3	4620	0.0142	0.1654
4	4759	0.0198	0.1855
5	4388	0.0171	0.1779
6	4444	0.0150	0.1456
7	4449	0.0148	0.1682
8	4275	0.0116	0.1710
9	4019	0.0098	0.1916
10	3636	0.0069	0.1986
Total	52.85 %	Average	0.0135

Table 5.6: Summary of order relation deviations for a particular realization in SIS.

Order Relation Deviations

Order relation deviations occurred in around 52 % of the points simulated with an average magnitude of less than 1.5 %. This means, on average, the cumulative probability values corresponding to each threshold were corrected by this amount. The maximum correction due to order relation was of 20 %. These corrections are within the range that is commonly seen in practice [43]. Hence, they are deemed acceptable and should not affect considerably the performance of the numerical models generated.

A summary of the order relation deviation for a particular realization is shown in **Table 5.6**.

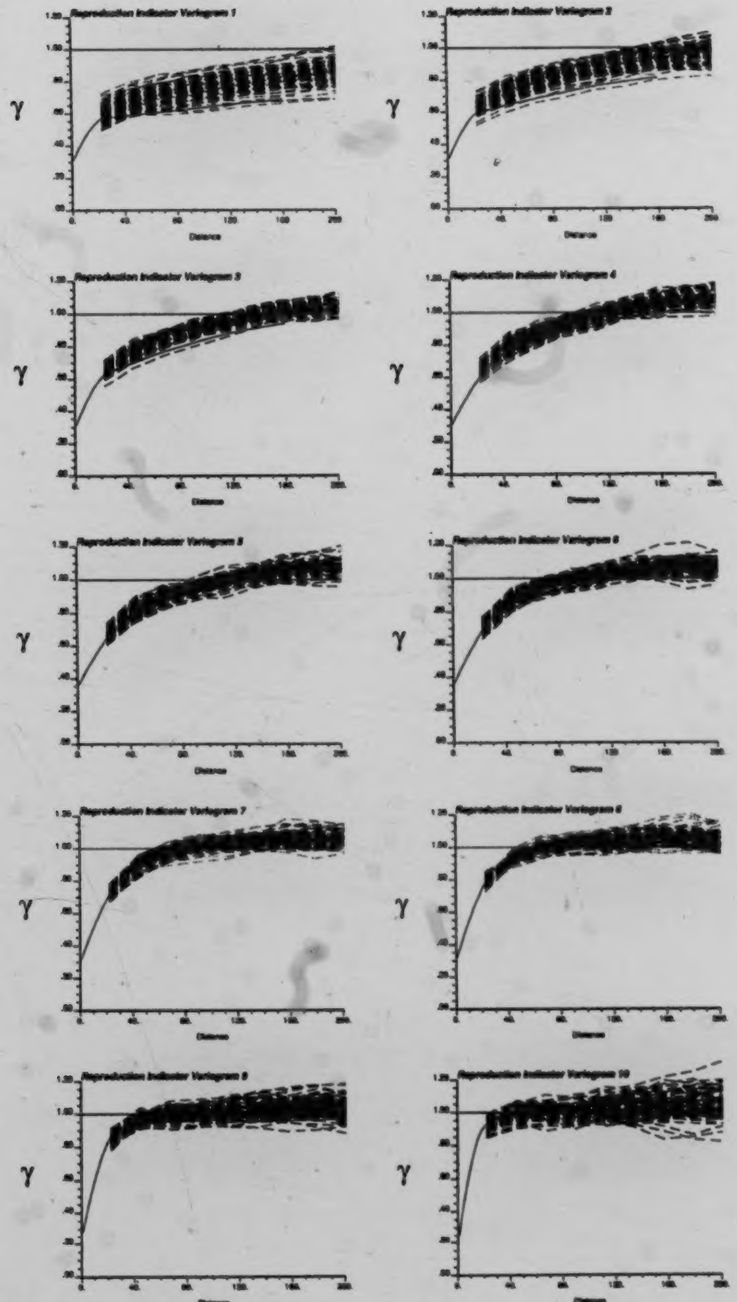


Figure 5.26: Indicator variogram reproduction for direction N30°W (**SIS**).

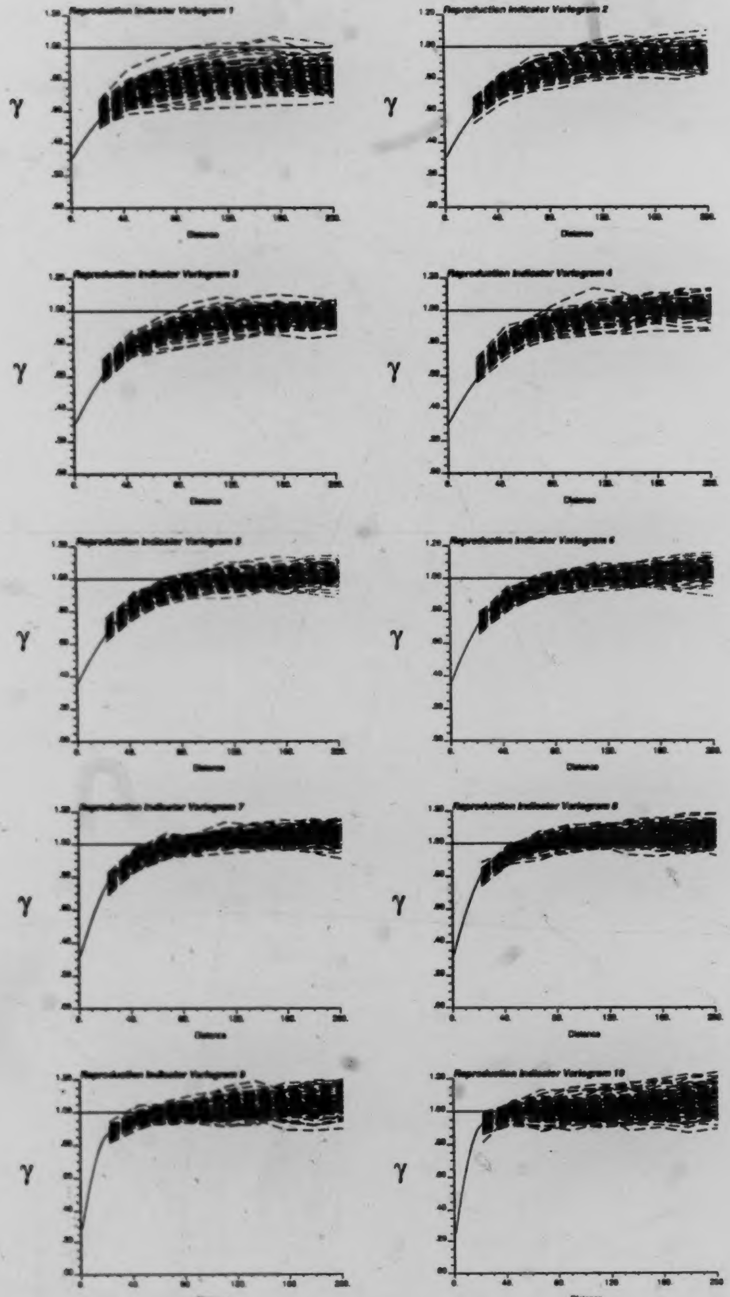


Figure 5.27: Indicator variogram reproduction for direction N60°E (SIS).

5.6 Assumption of Independence between Single-Point (DH Data) and Multiple-Point Information (BH Data)

The practical implementation of updating the indicator kriging probability with MP information under the assumption that both sources of information are independent was done with serious limitations due to the problem of order relations and consistency between the statistics.

Under this assumption, the updated probability can easily have a value over one. Furthermore, this value can be extremely high when a low threshold is considered. For example, assume the IK probability is 0.9, the MP probability is 0.9, and the global cumulative probability for that threshold is 0.1. The updated probability under the assumption of independence of both sources of information is: $P(\mathbf{A}|\mathbf{B},\mathbf{C}) = 0.9 \cdot 0.9/0.1 = 8.1$.

This property of the updating technique implies that large corrections are applied to satisfy the requirements of a cumulative distribution, when all the thresholds are estimated by IK and updated under this assumption, which introduces a severe bias in the resulting proportions.

5.6.1 Parameters

The same parameters described for the implementation of **SIS** were used in this case (see **Table 5.5**). Additionally to these, multiple-point statistics inferred from the blasthole data on the two benches above the ones simulated are used (see **Section 5.4**).

5.6.2 Validation of Results

Reproduction of Statistics

100 realizations were computed updating under the assumption of independence of the two sources of information. The implementation of this method implies severe order relation corrections that produce a bias, particularly for low thresholds. The effect of order relation deviations can be seen in **Figure 5.28**, where a bias in the histogram and q-q plot is evident.

The means and variances of the realizations also reflect this bias, as expected (**Figure 5.29**).

Another problem is due to the inconsistency of the distribution of both sources of data: the univariate distribution of drillhole data used as a representative distribution and blasthole data used to infer the MP statistics do not match exactly. The proportions below the ten thresholds are slightly different. For this reason, the bias in the MP information was corrected by replacing the cumulative probability from the reference distribution by the corresponding value from the univariate distribution of the data used to infer the multiple-point statistics.

Again, 100 realizations were computed with the results shown in **Figures 5.30** and **5.31**. The mean could not be reproduced.

Although the correction improves the result, the simulated models still show a strong bias with respect to the mean. The bias is due to the large order relation

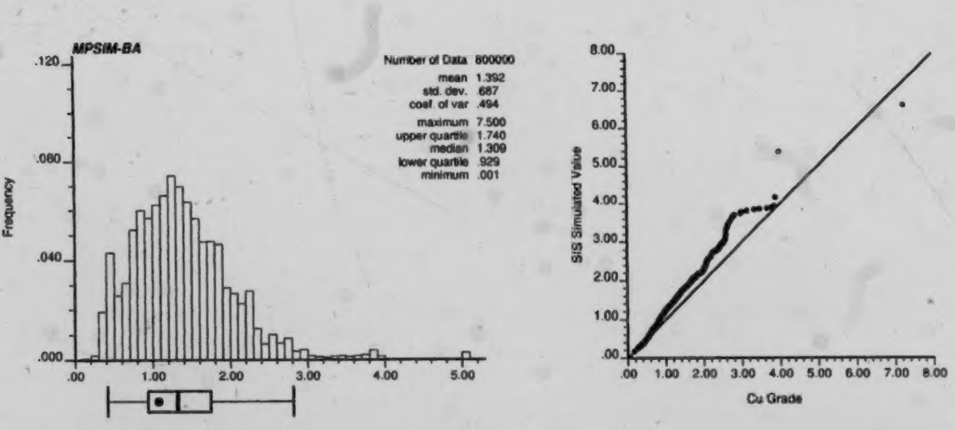


Figure 5.28: Histogram and q-q plot of all the simulated values (100 realizations) under the assumption of independence of the sources of information. The dot represents the mean from the reference declustered distribution.

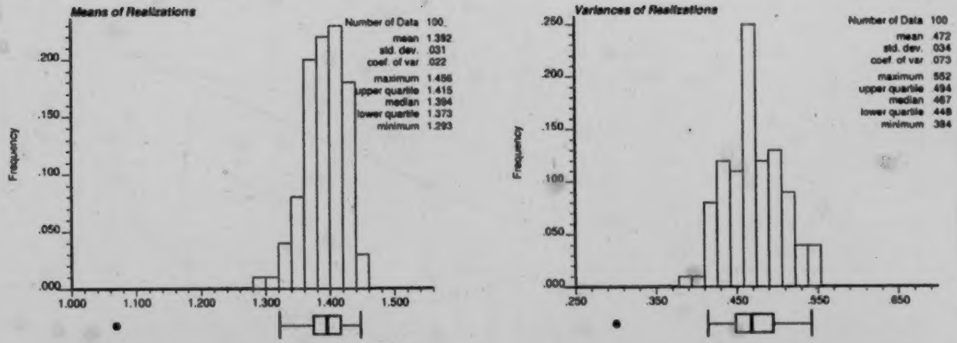


Figure 5.29: Histograms of the means and variances of the realizations obtained by updating under the independence assumption. The dots below the histogram represent the corresponding reference values.

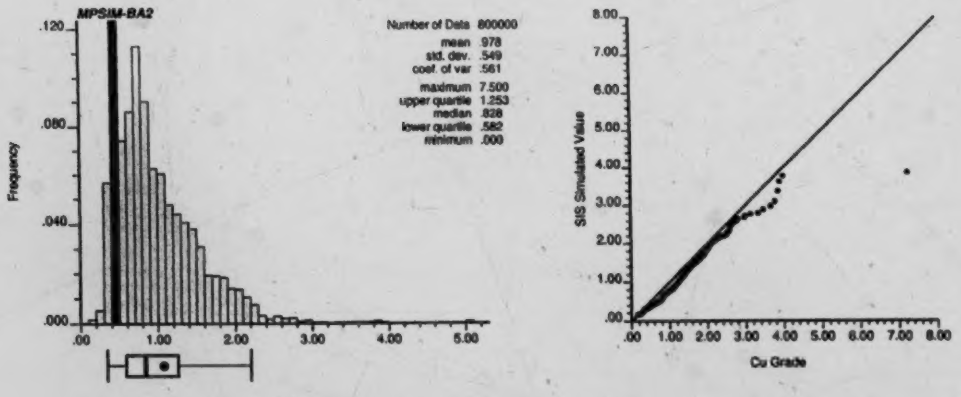


Figure 5.30: Histogram and q-q plot of all the simulated values (100 realizations) under the assumption of independence of the sources of information. The dot represents the mean from the reference declustered distribution.

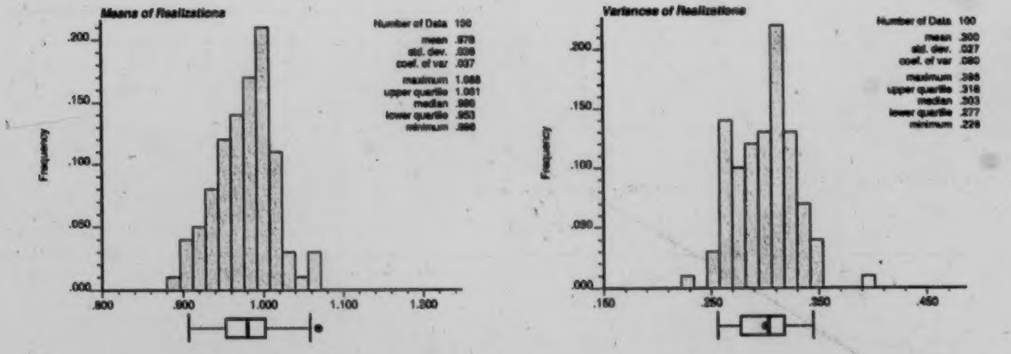


Figure 5.31: Histograms of the means and variances of the realizations under the assumption of independence of the sources of information. The dots below the histogram represent the corresponding reference values.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	4484	0.1653	4.6875
2	5028	0.0686	1.8496
3	5255	0.0788	1.2642
4	5206	0.0867	0.9427
5	4812	0.0682	0.5572
6	4445	0.0664	0.3711
7	4533	0.0606	0.3641
8	4243	0.0510	0.3761
9	4580	0.0276	0.4040
10	4539	0.0158	0.3207
Total	58.91 %	Average	0.0693

Table 5.7: Summary of order relation deviations for a particular realization, before correcting for inconsistency of univariate distributions, under the assumption of independence of the sources of information.

corrections to satisfy the requirements of a cumulative distribution. This will likely have consequences in the processing of the results.

Order Relation Deviations

Order relation deviations are significantly high for this algorithm. Summaries of order relation deviations for a particular realization before and after correcting for the inconsistency of univariate distributions are provided in **Tables 5.7** and **5.8**. Using a dynamic correction to fix the departure from the target proportions does not appear as a plausible solution. The sole existence of deviations of this magnitude renders the method unfit for practical applications.

It can be seen that the attempt to correct for the inconsistent probabilities below the thresholds obtained from the two sources of information worsens the order relation deviations.

Due to this problem, this method is discarded from further analysis and comparisons.

5.7 Assumption of Permanence of Ratios

The assumption of permanence of ratios has the advantage of generating an estimate that is always in the interval [0,1]. Combining this probability with the one obtained by indicator kriging does not generate large order relation deviations.

5.7.1 Parameters

The parameters used to update the IK probabilities with MP statistics under the assumption of permanence of ratios are the same than before (**Table 5.5**). MP statistics are inferred, as with the previous method, from the two benches above the ones being simulated.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	4420	1.1061	8.9115
2	5700	0.7532	3.7157
3	6164	0.5804	2.5376
4	6172	0.4829	1.7835
5	5974	0.3435	1.0661
6	6199	0.2538	0.6607
7	6130	0.1850	0.4449
8	5989	0.1320	0.4265
9	6194	0.0630	0.4129
10	6239	0.0373	0.3600
Total	73.98 %	Average	0.3703

Table 5.8: Summary of order relation deviations for a particular realization, after correcting for inconsistency of univariate distributions, under the assumption of independence of the sources of information.

5.7.2 Validation of Results

Reproduction of Statistics

The method is first applied disregarding the discrepancy between the univariate distribution of drillhole and the one of blastholes used to infer the multiple-point statistics. The reproduction of global statistics is shown in **Figures 5.32** and **5.33**. Results show again that a severe bias in these statistics stems from the mismatch between the proportions below the thresholds calculated from the univariate distributions of drillholes and blastholes.

The mismatch is corrected by using $P(\mathbf{A})$ obtained from the blasthole grade distribution. The new implementation results in a much better reproduction of the statistics. The tradeoff is an inflation of the variance of the realizations (**Figures 5.34** and **5.35**).

Maps of the first two realizations are shown in **Figure 5.36**. Comparing these maps with the ones obtained by `SIS` (**Figure 5.20**), the higher connectivity of highs and lows can be appreciated.

The reproduction of the reference distribution on a realization basis is presented in **Figure 5.37**.

Reproduction of Data Values

As before, the drillhole samples are assigned to the nodes in the grid. The same procedure for `SIS` is used and around 90 % of the samples are reproduced, with the other 10 % not assigned to a node because a closer sample was available (see **Figure 5.24**).

Reproduction of Indicator Variograms

The impact of adding multiple-point information to the models is reflected in the reproduction of the indicator variograms. A larger range is seen in most cases,

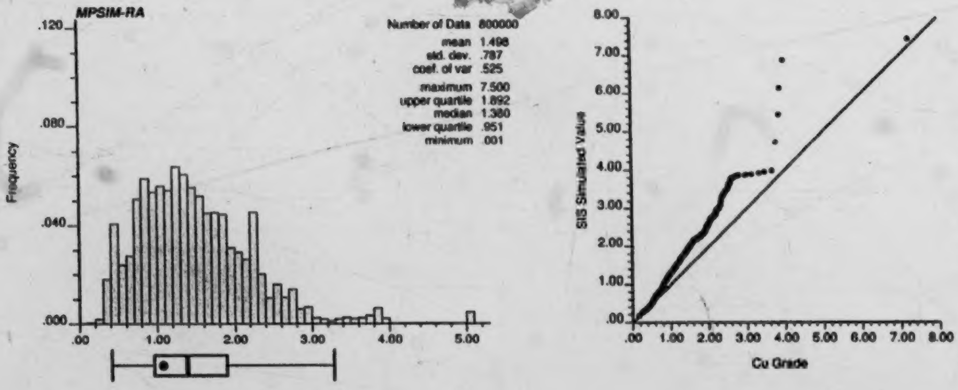


Figure 5.32: Histogram and q-q plot of all the simulated values (100 realizations) under the assumption of permanence of ratios. The dot represents the mean from the reference declustered distribution.

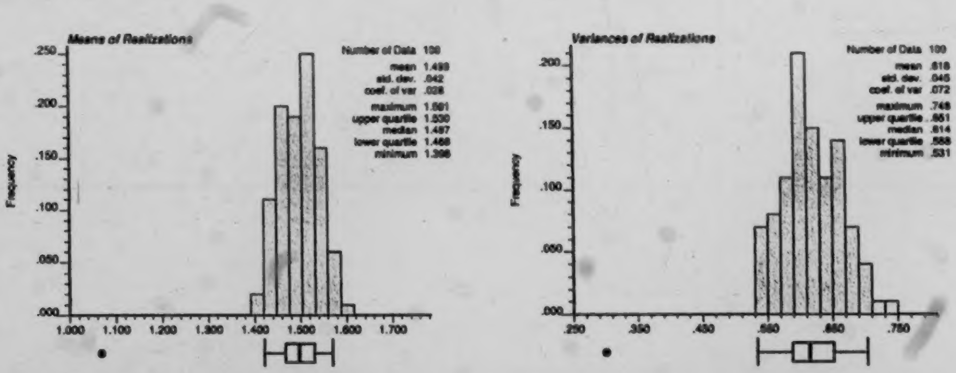


Figure 5.33: Histograms of the means and variances of the realizations under the assumption of permanence of ratios. The dots below the histogram represent the corresponding reference values.

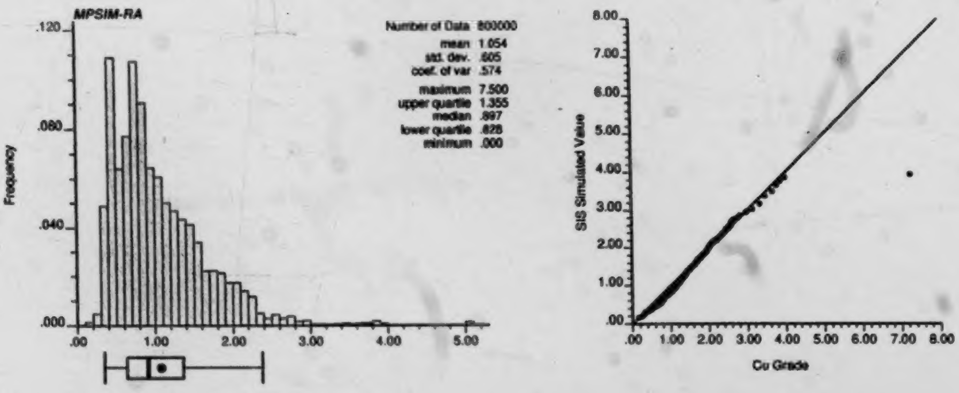


Figure 5.34: Histogram and q-q plot of all the simulated values (100 realizations) under the assumption of permanence of ratios. The dot represents the mean from the reference declustered distribution.

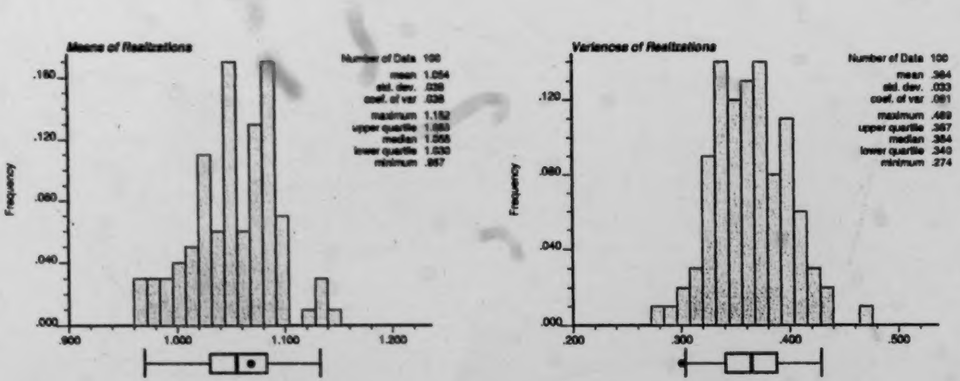


Figure 5.35: Histograms of the means and variances of the realizations under the assumption of permanence of ratios. The dots below the histogram represent the corresponding reference values.

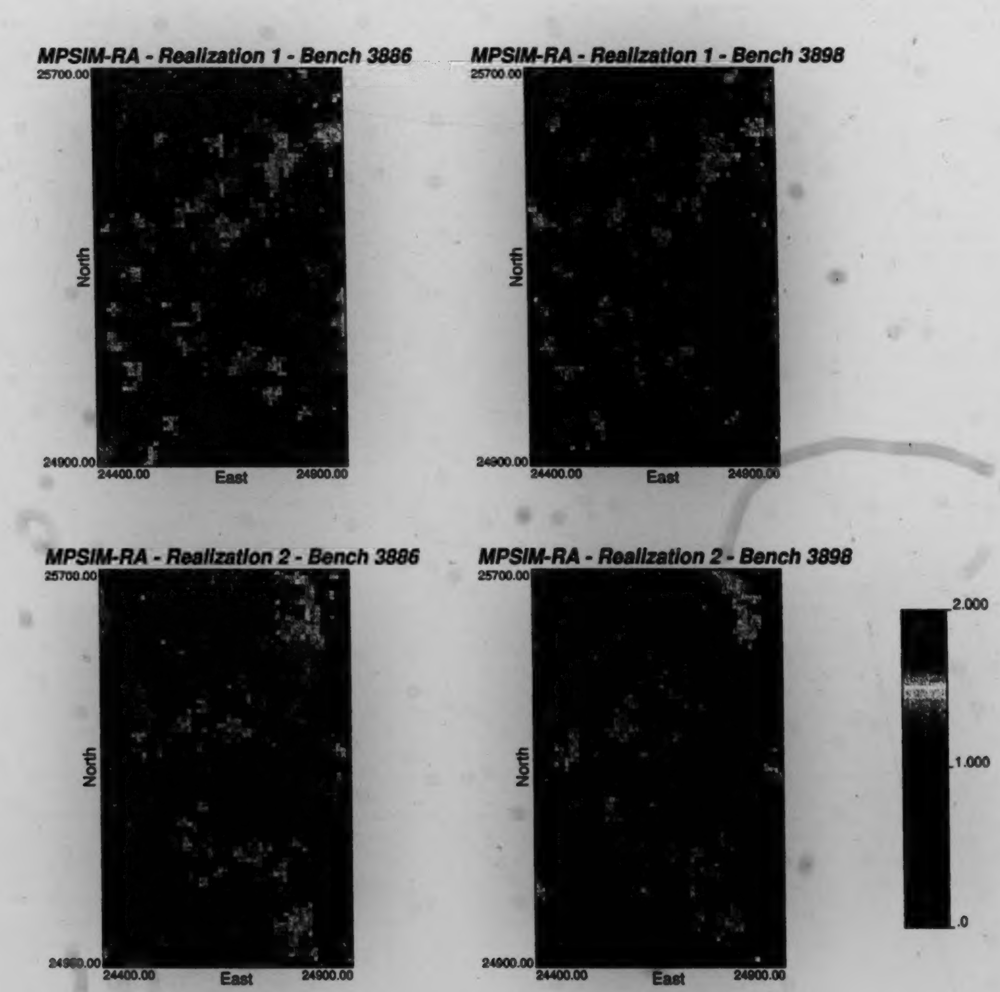

MPSIM-RA - Realization 1 - Bench 3886
MPSIM-RA - Realization 1 - Bench 3898
MPSIM-RA - Realization 2 - Bench 3886
MPSIM-RA - Realization 2 - Bench 3898
25700.00
24900.00
North
East
24400.00
24900.00
2.000
1.000
.0

Figure 5.36: Maps of the two benches for the first two realizations under the assumption of permanence of ratios.

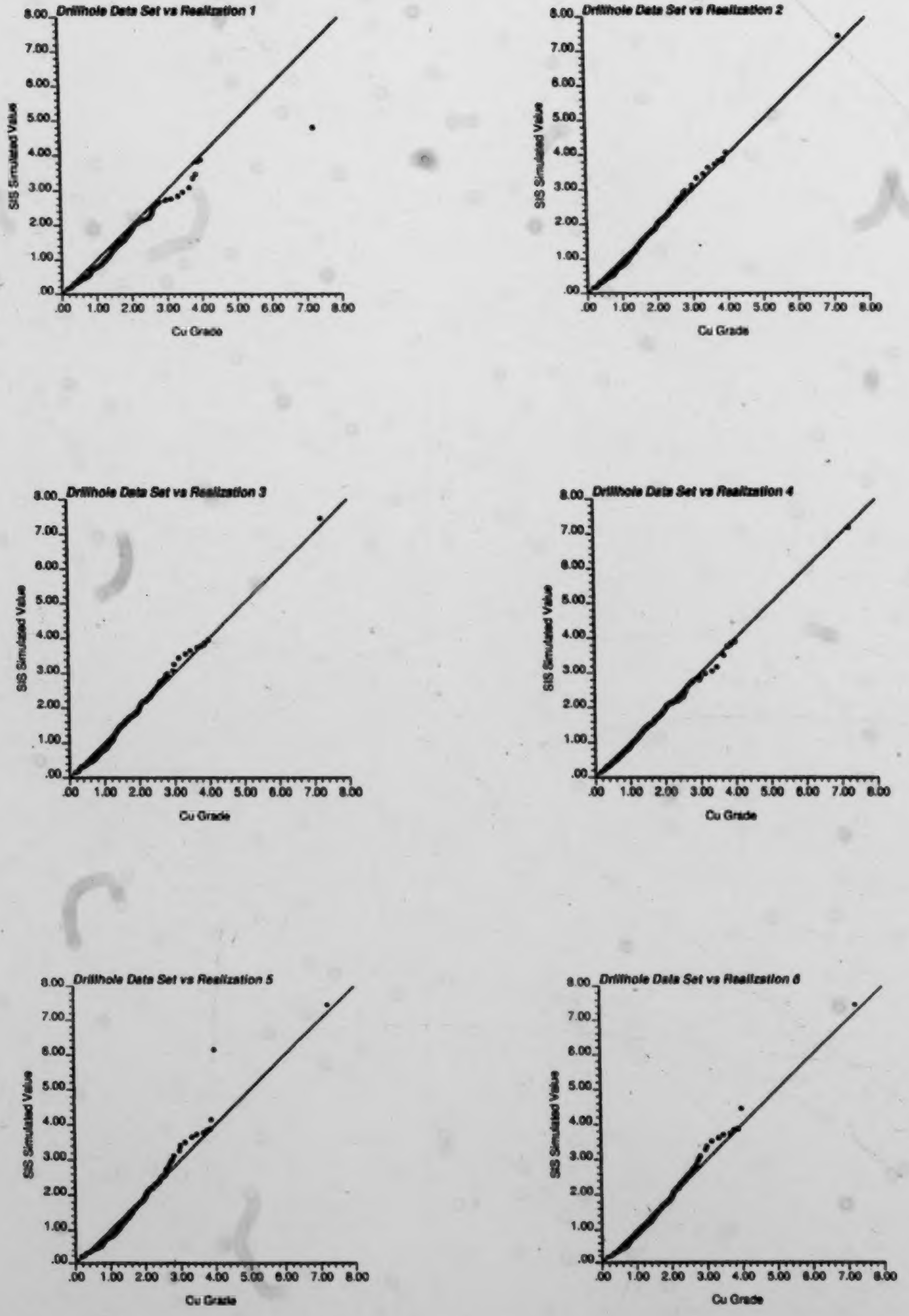


Figure 5.37: Q-Q plots of the reference distribution versus the distribution from the first six simulated models under the assumption of permanence of ratios.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	4713	0.0147	0.3884
2	5352	0.0141	0.3849
3	5832	0.0216	0.3639
4	5923	0.0247	0.3468
5	5563	0.0261	0.3034
6	5274	0.0264	0.3028
7	5330	0.0266	0.2881
8	5282	0.0286	0.3534
9	5098	0.0218	0.3274
10	4583	0.0284	0.3884
Total	66.19 %	Average	0.0233

Table 5.9: Summary of order relation deviations for a particular realization under the assumption of permanence of ratios, before correcting for inconsistency of univariate distributions.

which is consistent with results obtained by other researchers [42] (**Figures 5.38** and **5.39**).

Order Relation Deviations

Order relation deviations are relatively small, but slightly higher than in `SIS`. They should not affect the performance of the models. As presented in **Tables 5.9** and **5.10**, corrections are on average smaller than 2.5 %, with maximums reaching up to 40 %.

5.8 Multi-Gaussian Assumption

The multi-Gaussian assumption to approximate the redundancy between the two sources of information also provides reasonable estimates, with conditional probabilities generally not outside the interval [0,1]. Order relations do not generate major difficulties in its implementation.

5.8.1 Parameters

The parameters in **Table 5.5** are once again used to update the IK probabilities with MP statistics under the multi-Gaussian assumption.

5.8.2 Validation of Results

Reproduction of Statistics

Without correcting for the difference between the univariate statistics of drillhole and blasthole data sets, global statistics are poorly reproduced (**Figures 5.40** and **5.41**).

The use of the probabilities obtained from the blasthole dataset, from where MP statistics are inferred, results in a much better reproduction of the reference

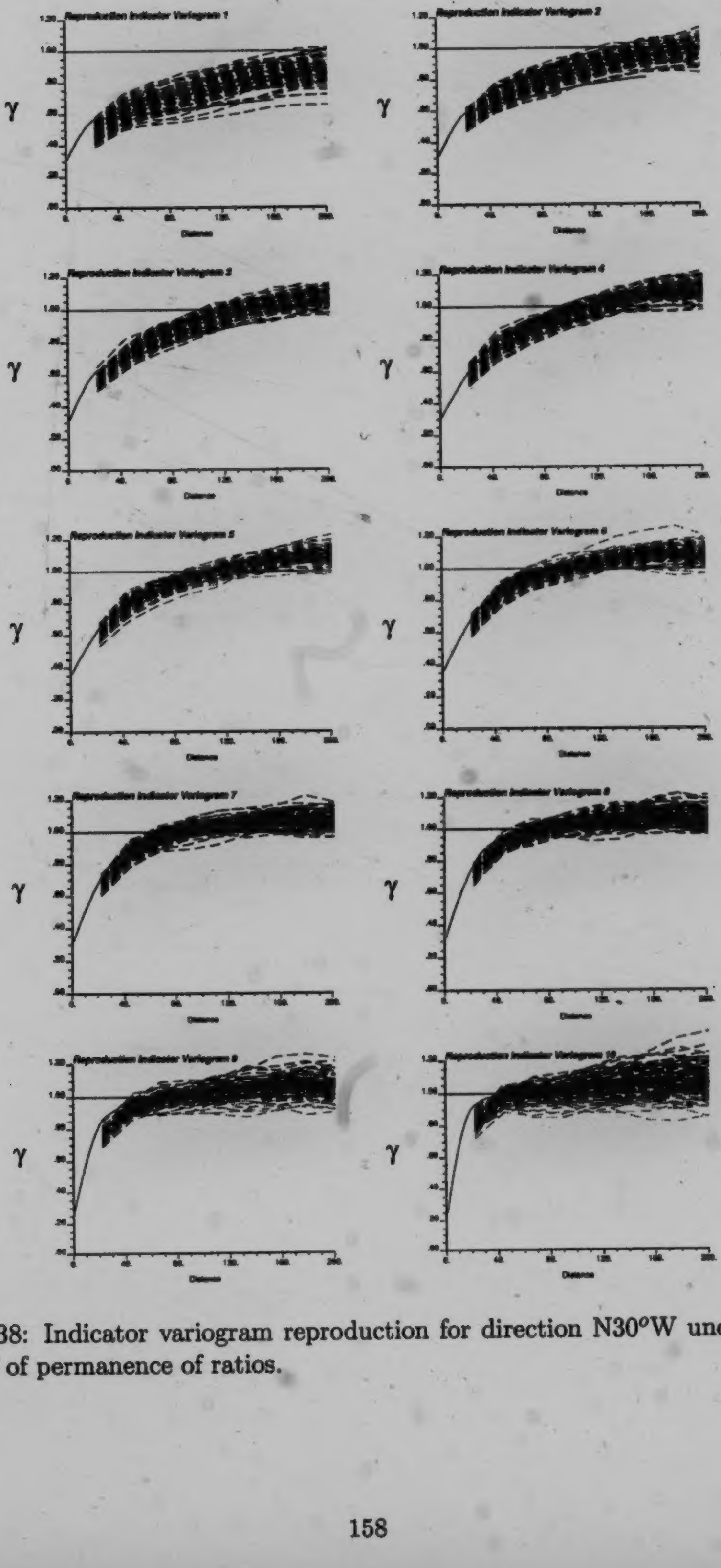


Figure 5.38: Indicator variogram reproduction for direction N30°W under the assumption of permanence of ratios.

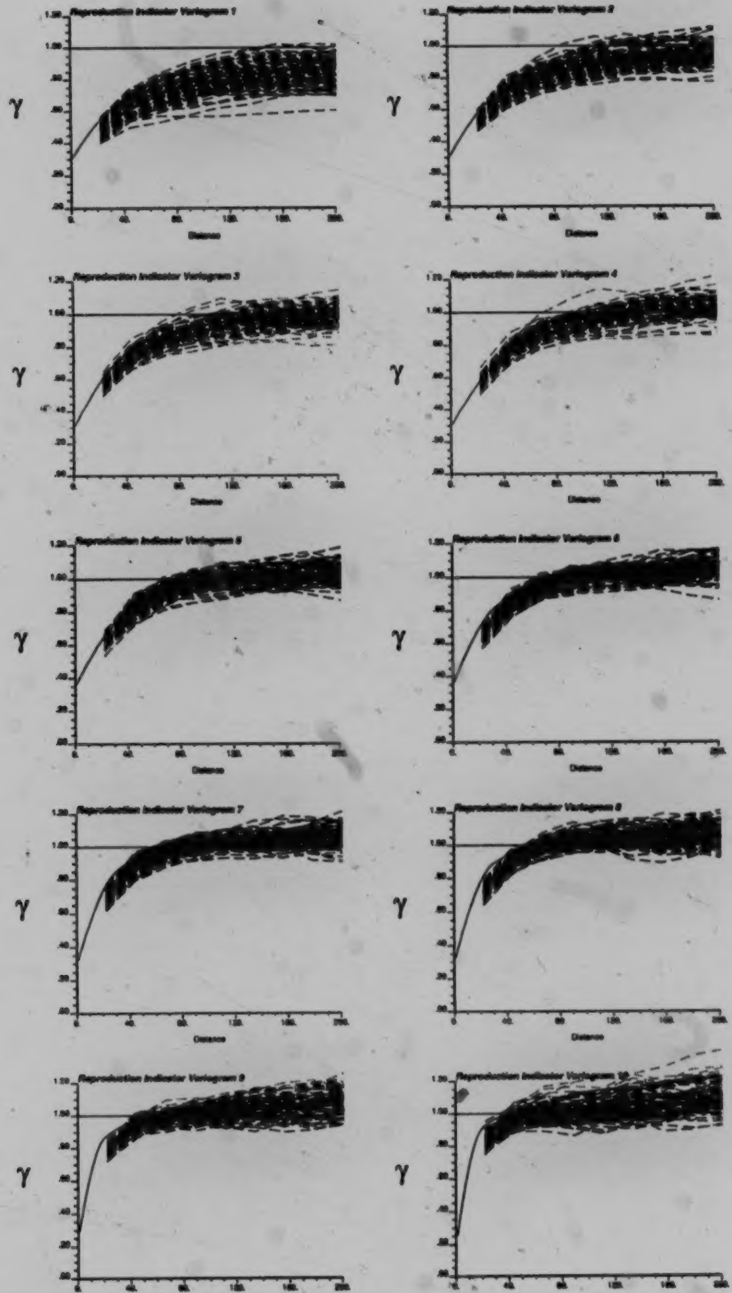


Figure 5.39: Indicator variogram reproduction for direction N60°E under the assumption of permanence of ratios.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	4284	0.0357	0.4065
2	4976	0.0306	0.3885
3	5852	0.0286	0.3529
4	6159	0.0276	0.3377
5	5999	0.0260	0.3533
6	6127	0.0231	0.2816
7	6017	0.0253	0.3442
8	5651	0.0215	0.3724
9	4776	0.0144	0.4065
10	4059	0.0086	0.3885
Total	67.38 %	Average	0.0244

Table 5.10: Summary of order relation deviations for a particular realization under the assumption of permanence of ratios, after correcting for inconsistency of univariate distributions.

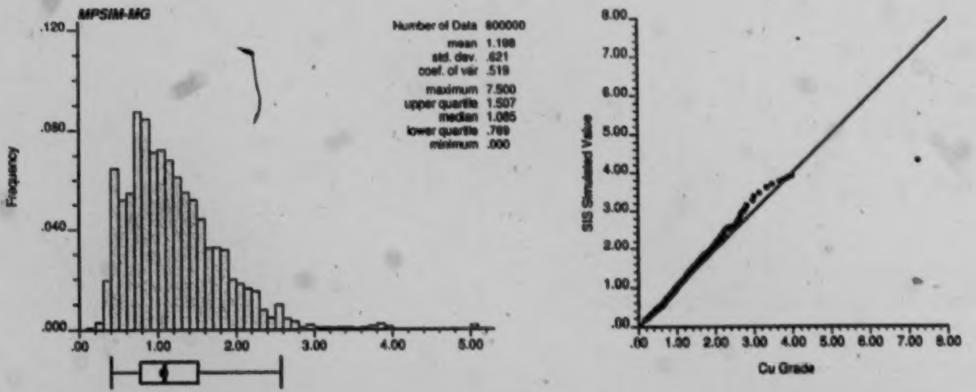


Figure 5.40: Histogram and q-q plot of all the simulated values (100 realizations) under the multi-Gaussian assumption before correcting for inconsistency between univariate distributions. The dot represents the mean from the reference declustered distribution.

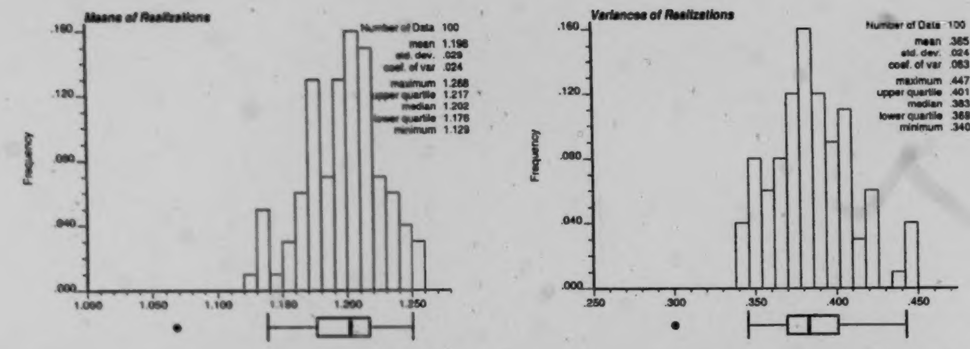


Figure 5.41: Histograms of the means and variances of the realizations under the multi-Gaussian assumption before correcting for inconsistency between univariate distributions. The dots below the histogram represent the corresponding reference values.

distribution (**Figures 5.42** and **5.43**). The mean of the simulated models appears slightly higher than the reference value, and the variance is correctly reproduced.

Maps of the first two realizations are shown in **Figure 5.44**. These maps do not show the high connectivity obtained through the assumption of permanence of ratios.

Q-Q plots of the first six realizations versus the reference distribution show the good reproduction of it (**Figure 5.45**).

Reproduction of Data Values

Data values are reproduced in the same manner than with the other methods. All data assigned to a node are correctly reproduced, and only the few ones that cannot be assigned because another available data is closer to the node where it was to be assigned, are not honored.

Reproduction of Indicator Variograms

The impact of this assumption on the reproduction of the indicator variograms is not evident. It appears as if a bit more variability was added to them, although this is only a conjecture (**Figures 5.46** and **5.47**).

Order Relation Deviations

Order relation deviations are quite mild. In fact, this method generates a lower number of deviations than SIS with a similar average magnitude (see **Tables 5.11** and **5.12**).

5.9 Sequential Gaussian Simulation

Sequential Gaussian simulation is implemented in this section. Gaussian methods are by far the most used. Their main disadvantage is that all multiple-point statistics

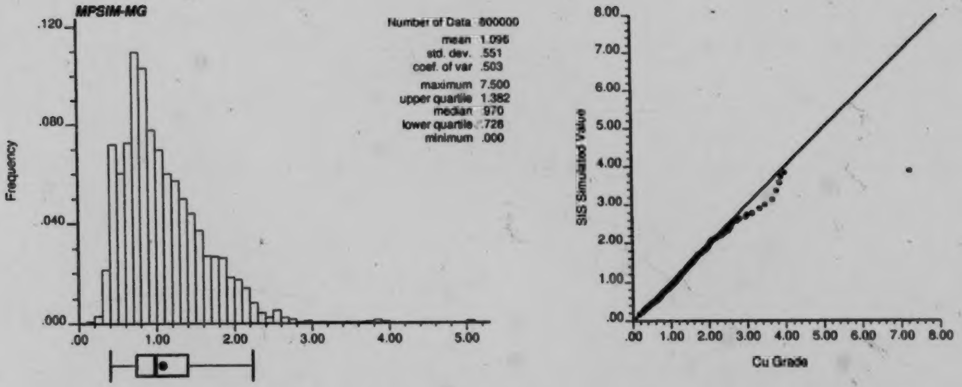


Figure 5.42: Histogram and q-q plot of all the simulated values (100 realizations) under the multi-Gaussian assumption after correcting for inconsistency between univariate distributions. The dot represents the mean from the reference declustered distribution.

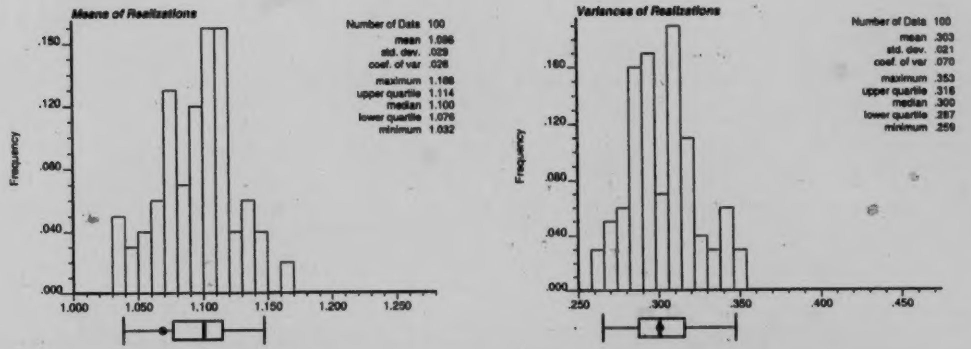


Figure 5.43: Histograms of the means and variances of the realizations under the multi-Gaussian assumption after correcting for inconsistency between univariate distributions. The dots below the histogram represent the corresponding reference values.

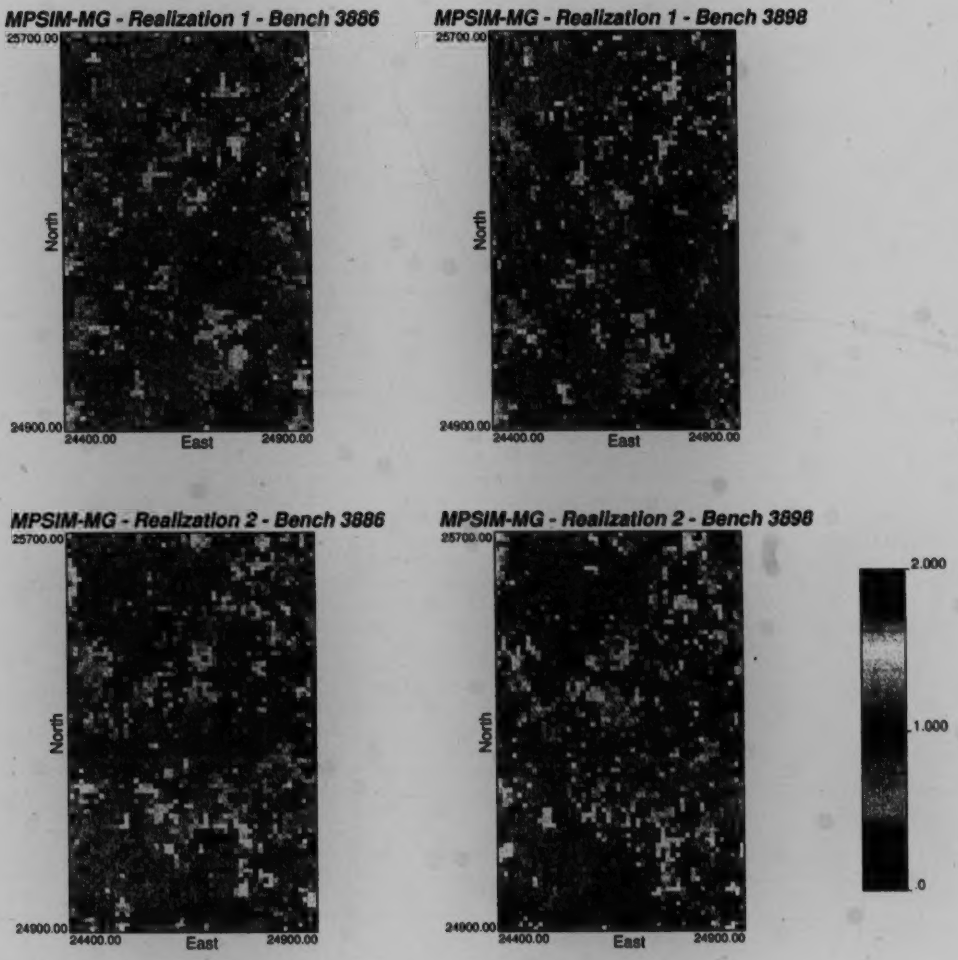

MPSIM-MG - Realization 1 - Bench 3886
MPSIM-MG - Realization 1 - Bench 3898
MPSIM-MG - Realization 2 - Bench 3886
MPSIM-MG - Realization 2 - Bench 3898
25700.00
24900.00
24400.00
24900.00
North
East
2.000
1.000
0

Figure 5.44: Maps of the two benches for the first two realizations under the multi-Gaussian assumption.

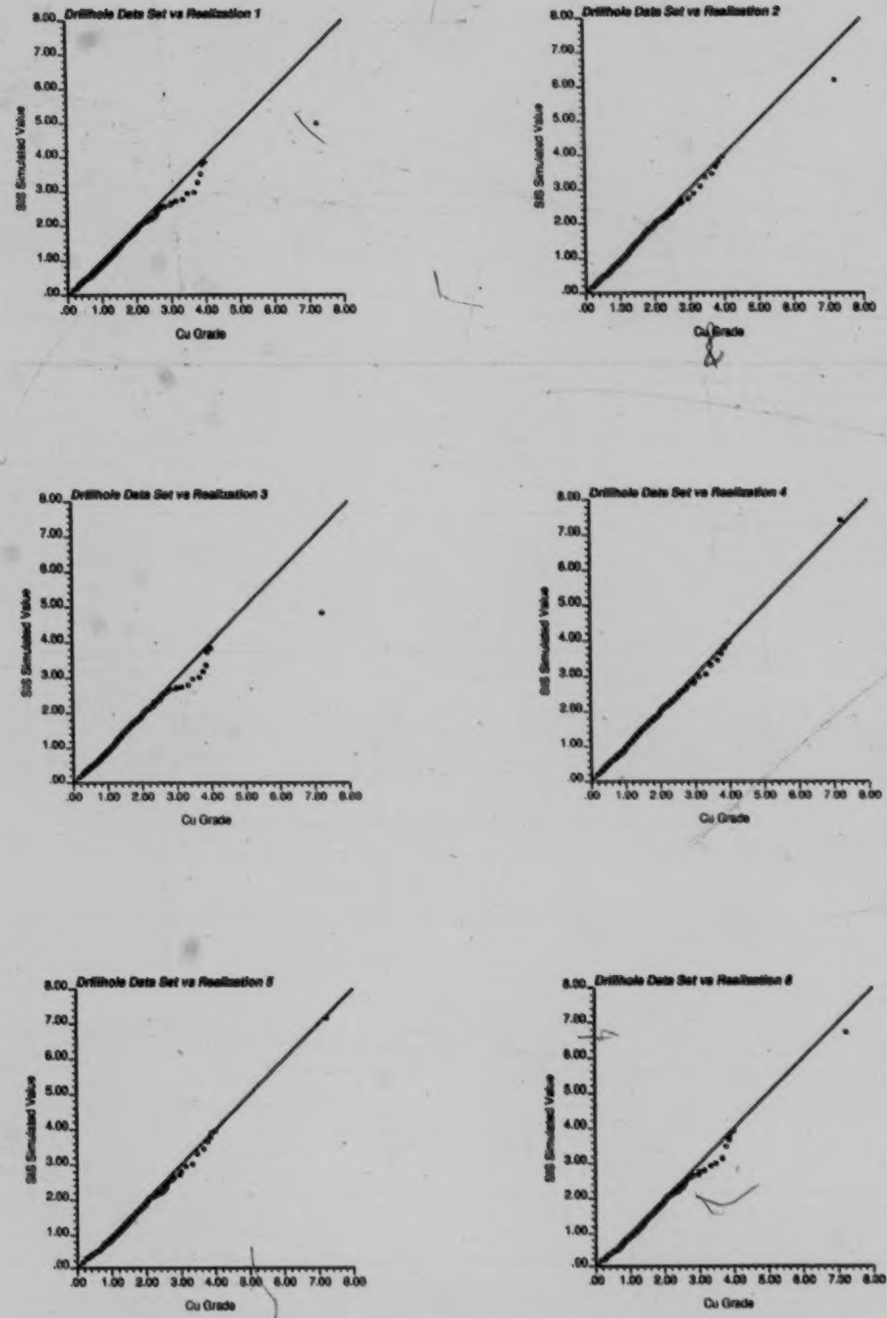


Figure 5.45: Q-Q plots of the reference distribution versus the distribution from the first six simulated models under the multi-Gaussian assumption.

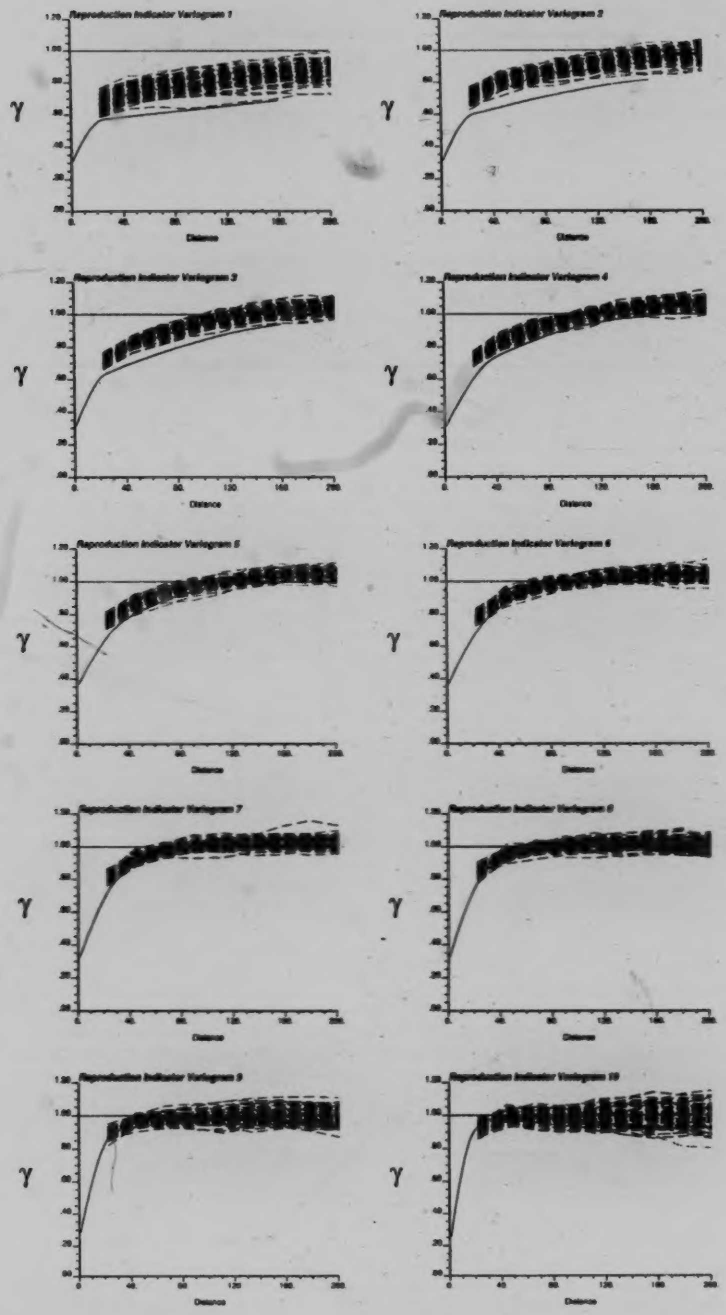


Figure 5.46: Indicator variogram reproduction for direction N30°W under the multi-Gaussian assumption.

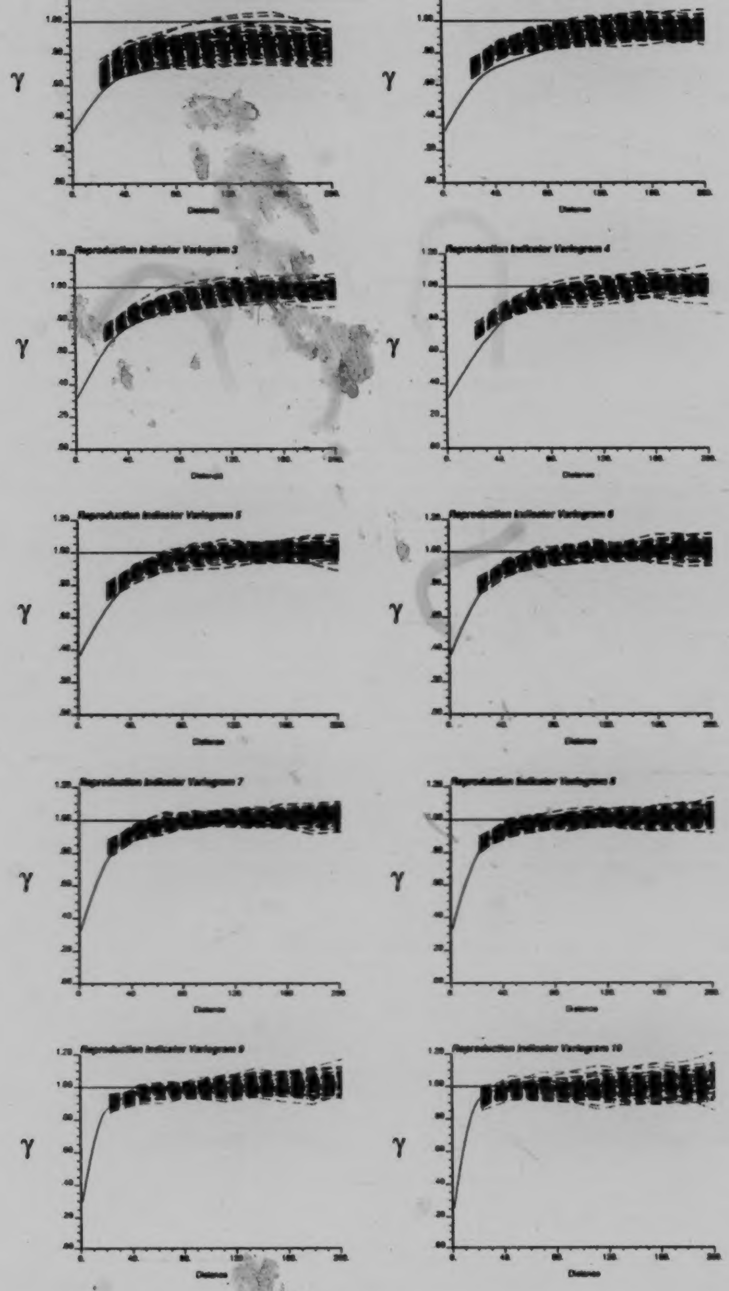


Figure 5.47: Indicator variogram reproduction for direction N60^{o}E under the multi-Gaussian assumption.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	4103	0.0111	0.2660
2	4573	0.0154	0.1846
3	4703	0.0187	0.2816
4	4379	0.0188	0.2438
5	3223	0.0192	0.2469
6	3079	0.0165	0.2867
7	2683	0.0167	0.1827
8	1865	0.0160	0.3722
9	1550	0.0115	0.3423
10	1023	0.0124	0.4297
Total	38.98 %	Average	0.0162

Table 5.11: Summary of order relation deviations for a particular realization under the multi-Gaussian assumption, before correcting for inconsistency of univariate distributions.

Threshold number	Number of corrections	Average deviation	Maximum deviation
1	1637	0.0137	0.2330
2	2367	0.0150	0.1893
3	3091	0.0166	0.3066
4	3468	0.0204	0.2217
5	3535	0.0174	0.2735
6	3542	0.0160	0.2863
7	3400	0.0150	0.2288
8	2950	0.0138	0.3586
9	2274	0.0093	0.3526
10	3029	0.0060	0.3803
Total	36.61 %	Average	0.0147

Table 5.12: Summary of order relation deviations for a particular realization under the multi-Gaussian assumption, after correcting for inconsistency of univariate distributions.

Nugget Effect	0.20
Structure 1	Spherical
Sill Contribution	0.15
Range N30°W	20.0
Range N60°E	60.0
Range Vertical	45.0
Structure 2	Exponential
Sill Contribution	0.70
Range N30°W	160.0
Range N60°E	105.0
Range Vertical	220.0

Table 5.13: Normal scores variogram model parameters.

are fixed, once the variogram model has been specified. There is also an increased loss of connectivity for extreme thresholds.

5.9.1 Normal Score Transformation

Transformation of the original grades to normal scores is required to calculate the variogram used in Gaussian simulation. This procedure is done by a standard graphical method [43]. Each sample value has now associated a normal value. A one-to-one relationship between the values in original units and the transformed values exists. In some cases "atoms" in the histogram may prevent this one-to-one relationship, that is, when many samples have the same value (typically zero or the detection limit of the sampling procedure). In these cases, despiking of the distribution would be required, however, given the characteristics of the global distribution of copper grades, this not deemed necessary.

5.9.2 Variogram of Normal Scores

The variogram of the normal scores is calculated with the same search and tolerance parameters used for the indicator variograms (**Table 5.2**). The final model is presented in **Table 5.13** and the experimental and fitted variograms in the three main directions of anisotropy are shown in **Figure 5.48**.

5.9.3 Parameters

The conventional program `sgsim` in GSLIB [43] is used to generate the Gaussian realizations. Data are assigned to nodes. The other parameters used in the simulation are presented in **Table 5.14**. Interpolation and extrapolation of the lower tail are done linearly. For the upper tail, a hyperbolic model with parameter $w = 1.5$ is used up to a grade of 7.5 %Cu. Maps of the first two renditions obtained are shown in **Figure 5.49**.

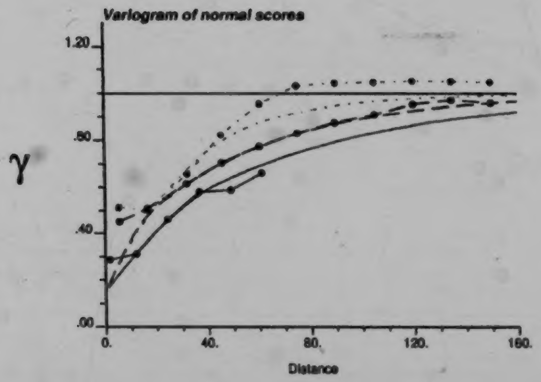


Figure 5.48: Normal scores variogram model. The continuous line corresponds to the vertical direction, the dashed line is in the N30°W direction, and the dotted line corresponds to the N60°E direction.

Random number generator seed	120574
Max. data and previously sim. nodes	24
Multiple-grid search levels	3
Maximum search radius horiz.	300.0 m
Maximum search radius vertical	150.0 m

Table 5.14: Simulation parameters.

5.9.4 Validation of Results

Reproduction of Statistics

The histogram and q-q plot of all the simulated realizations considered together are presented in **Figure 5.50**. Reproduction of the mean, variance, and quantiles of the reference distribution is satisfactory.

The histograms of means and variances calculated from each individual realization are shown in **Figure 5.51**. This graph shows the good reproduction of the histogram. The average variance of the realizations is smaller than the target value, which may be due to the conditioning data.

Plots comparing the distribution of grades for the first six realizations versus the target distribution are shown on **Figure 5.52**. Fluctuations occur, as expected.

Reproduction of Data Values

As with the previous methods, the data were assigned to grid nodes. The reproduction of data is then restricted to the samples that were actually assigned, while the samples for which another sample was available and closer to a grid node, were not honored.

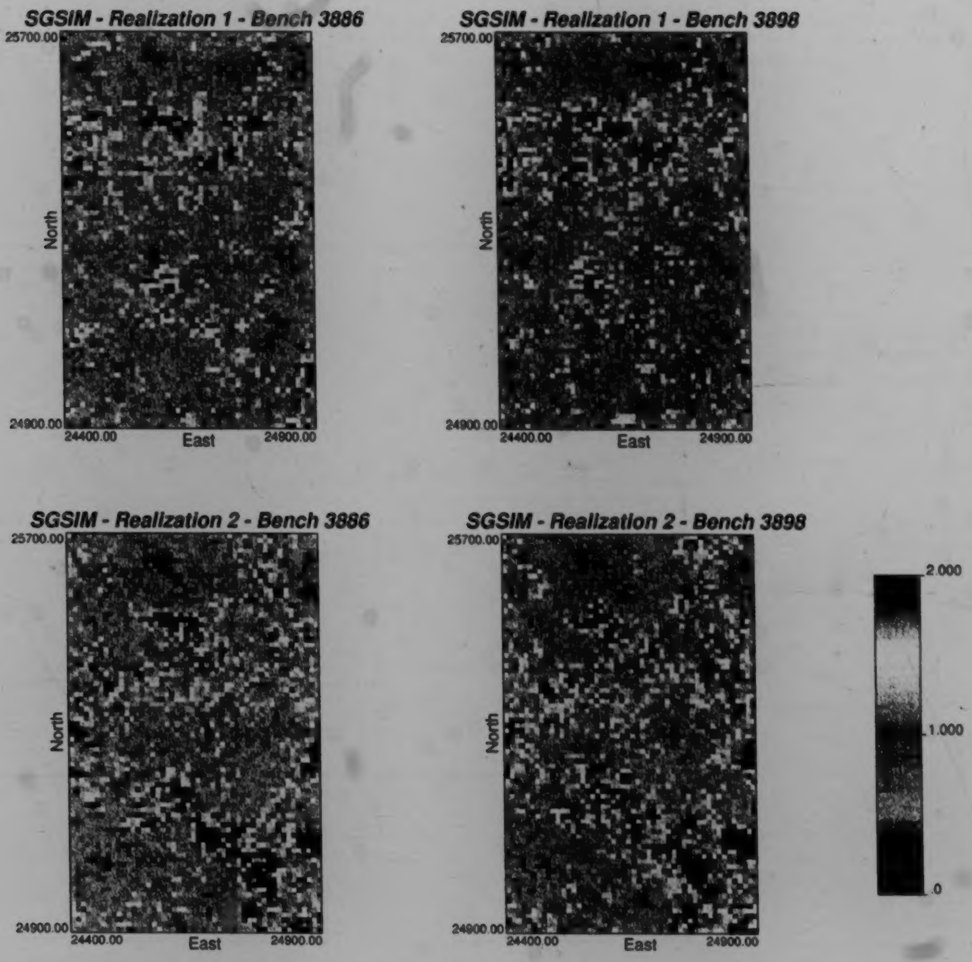

SGSIM - Realization 1 - Bench 3886
SGSIM - Realization 1 - Bench 3898
SGSIM - Realization 2 - Bench 3886
SGSIM - Realization 2 - Bench 3898
25700.00
24900.00
North
24400.00
East
24900.00
2.000
1.000
0

Figure 5.49: Maps of the two benches for the first two realizations using sequential Gaussian simulation.

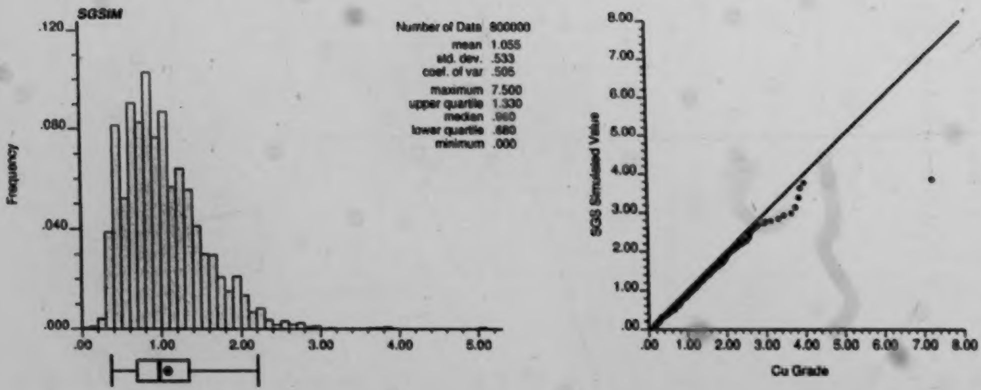


Figure 5.50: Histogram and q-q plot of all the simulated values by SGS (100 realizations). The dot represents the mean from the reference declustered distribution.

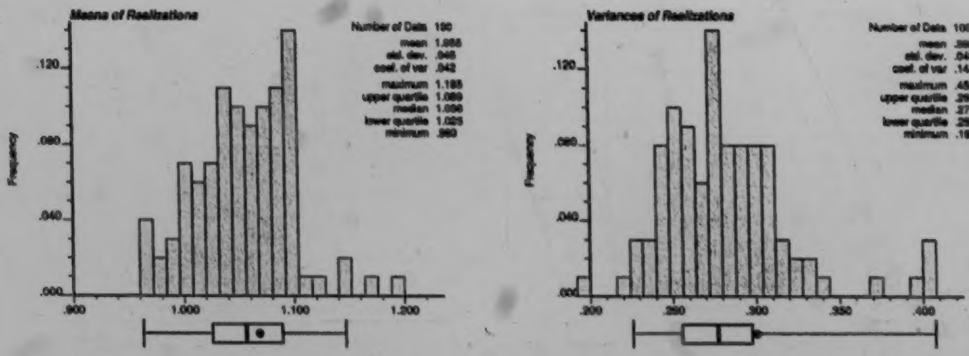


Figure 5.51: Histograms of the means and variances of the realizations by SGS. The dots below the histogram represent the corresponding reference values.

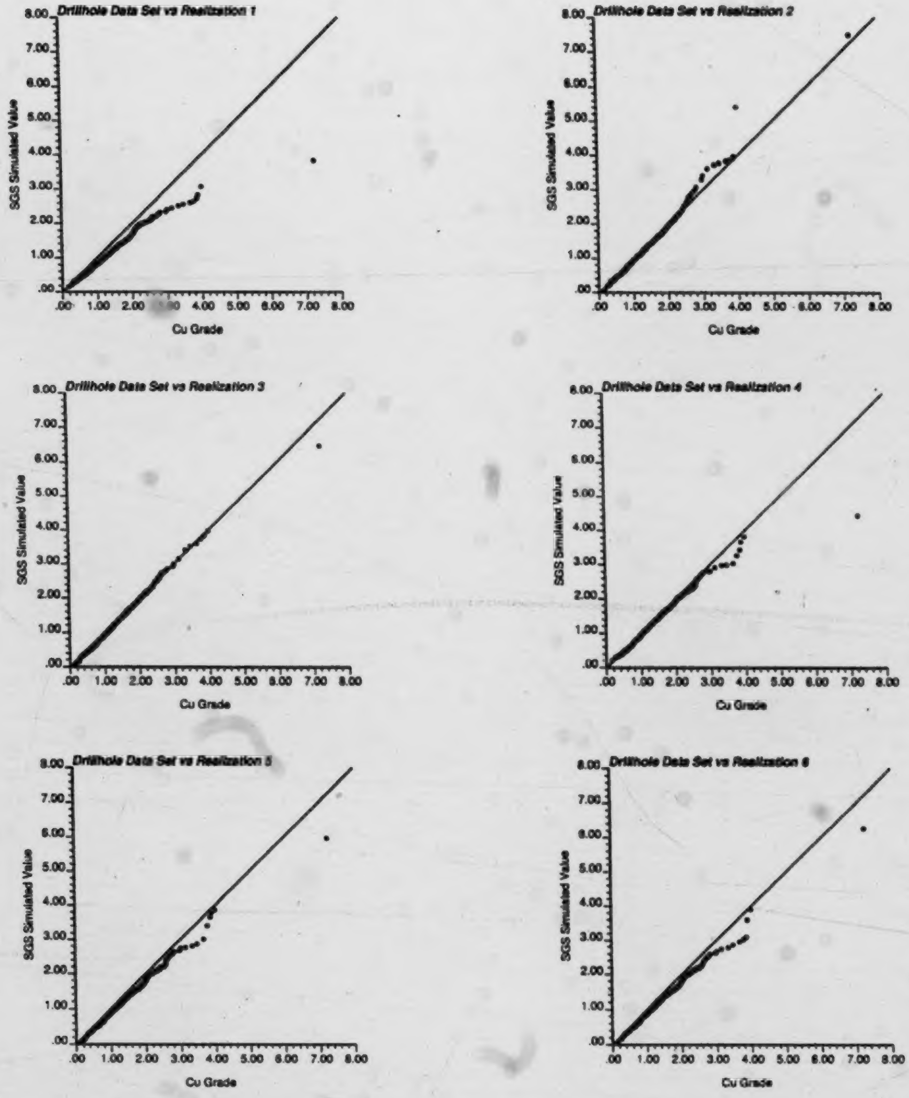


Figure 5.52: Q-Q plots of the reference distribution versus the distribution from the first six simulated models by SGS.

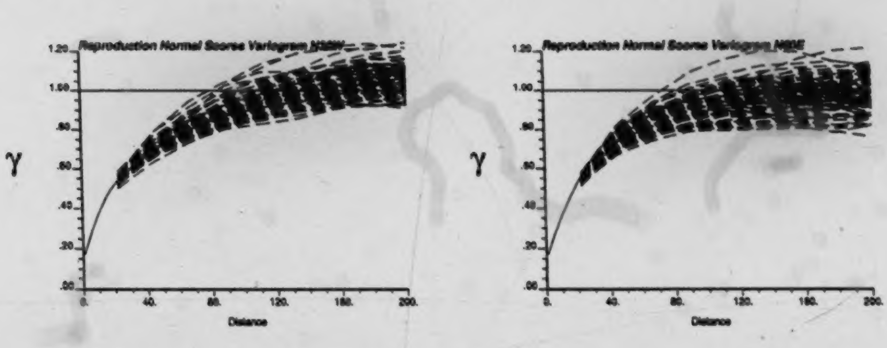


Figure 5.53: Variogram of normal scores reproduction for directions N30°W and N60°E (SGS).

Reproduction of Variogram of Normal Scores

For every realization, the variogram of normal scores was computed and compared to the model used in the simulation. **Figure 5.53** shows the reproduction. A slight bias can be seen in the N60°E direction.

5.10 Comparison of Results

5.10.1 Statistical Performance

To measure performance, the simulated models are compared with the available blasthole data kept for validation, for the two benches simulated. For each realization, the blasthole data are compared with the closest nodes in the simulated model, and the correlation coefficient is calculated. A histogram of these correlation coefficients summarizes the performance of indicator simulation and the other methods to predict the short term information provided by the blasthole data.

Models generated considering only two-point statistics via sequential indicator simulation gave an average correlation close to 0.30. The ones generated with sequential Gaussian simulation correlated better with the blastholes, the average coefficient of correlation was 0.33. Updating the IK probabilities with multiple-point statistics improved the average correlation to 0.35 under the assumption of permanence of ratios. The multi-Gaussian assumption performed poorly (**Figure 5.54**).

The improvement obtained by adding multiple-point statistics is significant. Considering indicator methods, the correlation coefficient goes up from 0.30 to 0.35 under the permanence of ratios assumption. Interestingly, Gaussian simulation outperforms the conventional indicator method, but still, the proposed method to integrate multiple-point statistics shows an even better performance. The assumption of permanence of ratios appear to be a robust way to integrate additional information into the indicator framework.

5.10.2 Mine Planning Performance

A second approach to measure performance is to consider the accuracy of the methods in predicting quantity of metal. Using the blasthole information kept for valida-

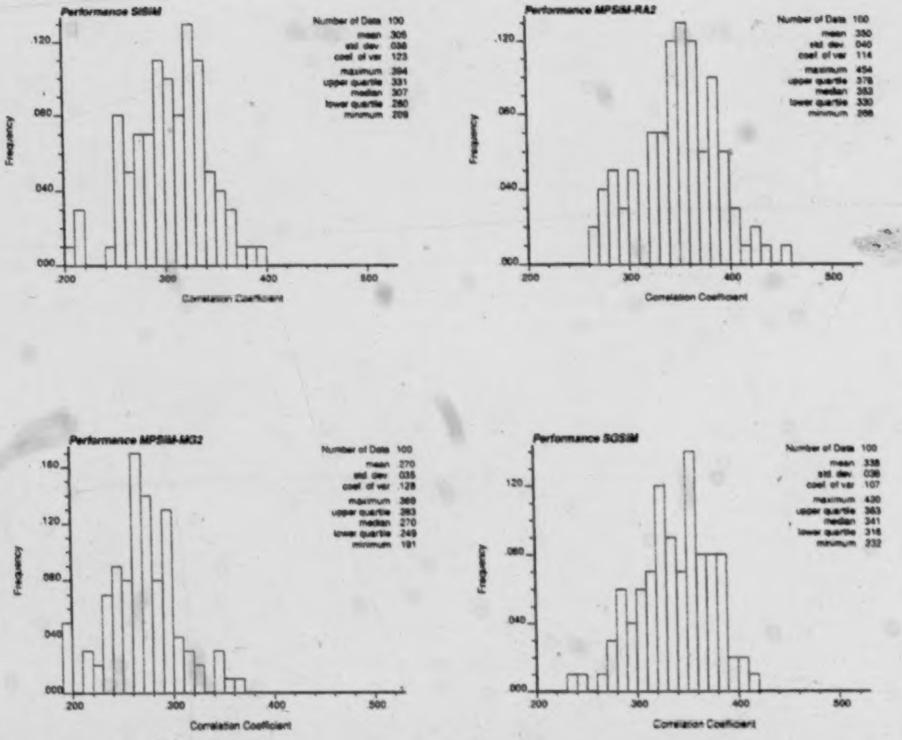


Figure 5.54: Histograms of correlation coefficients between the blasthole data and the closest simulated value, over 100 realizations. Top left: sequential indicator simulation; Top right: updating under the assumption of permanence of ratios; Bottom left: updating under the multi-Gaussian assumption; Bottom right: sequential Gaussian simulation.

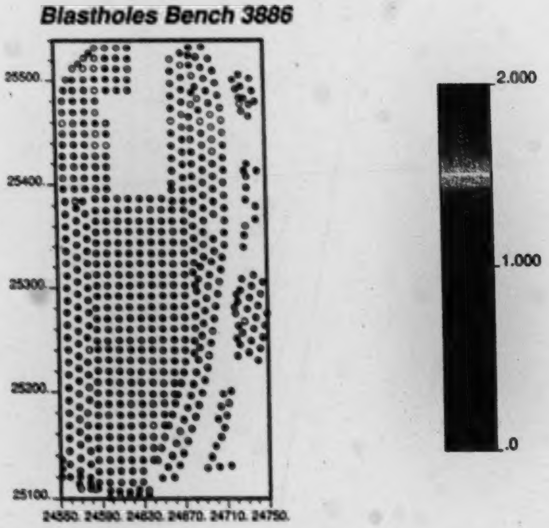


Figure 5.55: Area considered for calculation of quantity of metal. Blasthole data for the bench 3886 are shown.

tion, that is, the blastholes in benches 3886 and 3898, a map that will be considered the truth is built by ordinary kriging. The data are very densely located, hence the smoothing effect of kriging is not a concern. The quantity of metal is calculated considering blocks of 10 by 10 by 12 m^3, and a cutoff grade of 1.0 %Cu. The volume is restricted to the area where blasthole data are available, that is, only data within the volume defined by North coordinates between 24550 and 24750, and East coordinates between 25100 and 25540, are used (**Figure 5.55**). The performance of the proposed methods showed an improvement with respect to the conventional sequential indicator simulation approach of almost 3 % in terms of the mismatch with the truth obtained through kriging. The multi-Gaussian approach to assess the relationship between single and multiple-point information performed poorly, increasing the mismatch of the conventional technique. In this case, the error goes from -5.68% for `SIS` to -10.43% for the multi-Gaussian approximation. The simulation considering multiple-point statistics under the assumption of permanence of ratios also outperformed (although marginally) sequential Gaussian simulation. This corroborates the results in the previous section.

The experimental variogram is calculated and modelled for the principal directions of anisotropy (**Figure 5.56**).

The expected quantity of metal calculated over 100 realizations of sequential indicator simulation, updating under the permanence of ratios assumption, and updating under the multi-Gaussian assumption are compared with the true quantity of metal from ordinary kriging (**Table 5.15**).

5.10.3 Conclusions

The results indicate that better performance should be expected when incorporating multiple-point statistics into the simulation method. From the techniques proposed

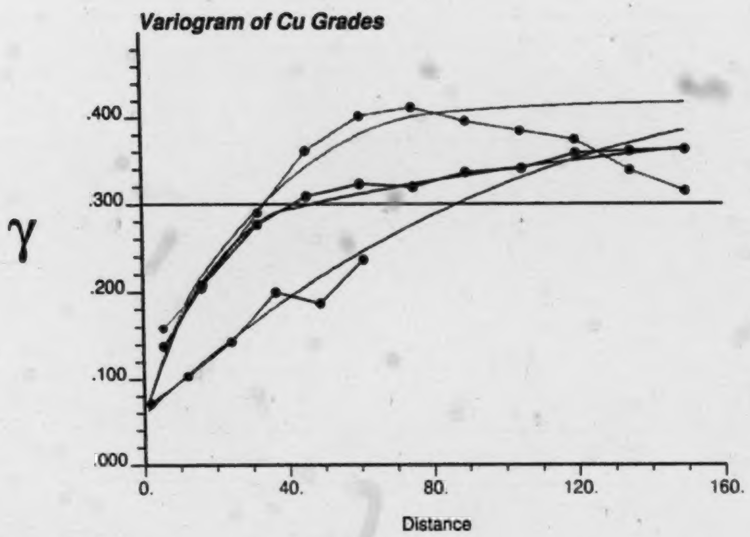


Figure 5.56: Experimental variogram of Cu grades and model used for ordinary kriging.

Method	Tons. Cu	Mismatch	Error %
Kriging (truth)	55274		
SIS	52036	-3238	-5.86
Perm. of Ratios	53679	-1595	-2.89
Multi-Gaussian	49510	-5764	-10.43
SGS	53614	-1660	-3.00

Table 5.15: Expected quantity of metal based on the different methods, compared to the "truth" computed by ordinary kriging of the blastholes.

to update the IK probability, the integration under the assumption of permanence of ratios appears as the best one. In this case study, it provides an estimate of the quantity of metal that is 3% closer to the true value than SIS, and marginally closer to the quantity of metal obtained with sequential Gaussian simulation.

The assumption of multi-Gaussianity to integrate the IK and multiple-point probabilities does not improve the estimation of quantity of metal.

This study suggests that multiple-point statistics extracted from data can be used to improve the numerical models built for medium and long term planning. However, abundant information is required in order to obtain reliable estimates of the probabilities of multiple-point events.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Since their first introduction more than ten years ago, the use of multiple-point statistics in simulation has been an active area of research in geostatistics. All applications have been based on the extraction of multiple-point statistics from training images and the focus has been put on the simulation algorithm used to reproduce them.

This research tries to approach this problem by privileging the use of data, instead of training images. These statistics are used to add quantitative information into the conventional sequential indicator simulation framework.

A first approach developed consists of a hierarchical simulation to account for the multiple-point information. This approach presents several challenges that can be further explored. Practical implementation has shown that the first approach produces artifacts that are undesirable and invalidate their application.

A second approach is to use Bayes' law to update the probability obtained by indicator kriging with some multiple-point information. The problem of compatibility of the two-point statistics, provided by the variogram or covariance, and multiple-point statistics, calculated as the probability of a node to be below a threshold given a specific arrangement of multiple values in space, can be resolved by considering some assumption about the redundancy of the two types of information. This second approach provides favorable results improving the performance of the numerical models for medium and long term planning.

The main conclusions that can be extracted from this research follow. Many of them can be seen as open paths to further research.

6.1.1 Incorporating Multiple-Point Runs in Geostatistical Simulation

The algorithm proposed to account for runs above and below several thresholds, in multiple directions was based on the simple idea of proceeding hierarchically from the highest threshold to the lowest, eroding the field.

The following ideas were developed in **Chapter 3**:

1. Runs in multiple directions must be honored for each threshold.

2. Due to the nesting property of runs, runs simulated at a higher threshold will constrain the domain for the subsequent lower thresholds, and can be considered as conditioning information.
3. The definition of a function to select the nodes to be switched (eroded) is required.
4. The selection function used can be considered as greedy, from a numerical optimization perspective, and pushes the histograms of runs close to the targets as fast as it can.
5. Convergence to the target statistics is not ensured by any means. Experimental results showed that satisfactory values could be obtained by defining an appropriate set of parameters for the selection function.
6. The appropriateness of these parameters cannot be generalized; they were found by trial and error.
7. Although the target statistics were reasonably reproduced, the algorithm generates undesired artifacts that make the realizations useless for further analysis.
8. The use of alternations to improve convergence appears as a possible solution to the defects of the algorithm as it stands.

Notwithstanding the apparent bad results, several theoretical developments were made during the consideration of this method. The expected probability of a run of a given length can be calculated in all generality if the spatial law of the random variable is known. This result allows the prediction of the occurrence of runs under the multi-Gaussian assumption.

The expected number of runs for a constant variogram model and different algorithms was also looked into, as presented in **Appendix B**. Very stable results for the expected number of runs and unstable ones for the variance of this value were noticed and could be further investigated.

6.1.2 Updating the Indicator Kriging Probability with Multiple-Point Statistics

The approach developed in **Chapter 4** considers the simplification of the expression of the conditional probability calculated as an indicator at an unsampled location, given several sources of information.

In general, the relationship between the sources of information is unknown. Some approximation must be made in order to allow its calculation. In this research, the relationship could be quantified if cross covariances between single-point and multiple-point events were known. This is what makes the use of multiple-point statistics a difficult matter: these cross-covariances are extremely hard to infer, and the modelling to ensure a positive definite function is rather complicated. Alternative approaches must be taken. The simplification of the redundancy terms is done by assuming some relationship between the sources of information.

The indicator kriging estimate and the probability of having a value below the threshold given a spatial arrangement of indicator values in a predefined pattern, are first considered independent. This entails a great simplification of the expression to obtain the conditional probability at an unsampled location. However, practical implementation of this method showed that large order relation corrections were required to keep the updated probabilities within the permitted range [0,1]. These large departures are acceptable when looked from the Bayesian perspective, but are unacceptable in the implementation of the algorithm. The method is deemed theoretically correct, but practically inapplicable.

An alternative assumption to integrate the two sources of information is the permanence of ratios. It means that the incremental information provided by one source of information is constant before and after knowing the information provided by other sources. It performs well in the spatial context presented in this research.

A final method to integrate multiple sources of information is proposed by means of the multivariate Gaussian assumption. The relationship between different sources of information is approximated as if the variable was multi-Gaussian. The theoretical derivation of this assumption is presented with unexpectedly poor performance in the case study.

The following points were developed in **Chapter 4**:

1. Updating of IK probability using MP statistics does not necessarily require a knowledge of the spatial law.
2. Assumptions can be made regarding the relationship between the different sources of information.
3. Three assumptions were developed for the purpose of incorporating MP statistics in a sequential indicator context. The description of the updating techniques has been made for the general case, where several sources of information are considered:
    - Independence between the sources of information. This appears as an unrealistic assumption in the context of spatial simulation. Its practical implementation carries important limitations due to order relation deviations.
    - Permanence of ratios. It entails that the incremental information provided by one source is independent of the other sources of information. This assumption is not as strong as the assumption of full independence between the sources of information, however it is not clear what is implicitly assumed. Nevertheless, its performance showed the best results among the methods proposed.
    - Multi-Gaussian approximation to assess the redundancy between the sources of information. What appeared to be the most realistic approach, in the sense that actual redundancy was being estimated as if the variable was multi-Gaussian, ended up performing poorly.

A case study using real data from an operating mine is presented in **Chapter 5**. Care has been taken to provide details of all the steps involved in the implementation of the techniques. The realizations are checked for data honoring, histogram, and indicator variograms reproduction. The indicator methods are implemented along with sequential Gaussian simulation, the most widely applied method for simulation. The performance is measured from two different points of view:

- A statistical measure of performance is considered by computing the correlation coefficient between blasthole data kept for validation and not used for inference or estimation, and the simulated numerical realizations of the variable by sequential indicator simulation and considering multiple-point statistics to update the IK probability under the different assumptions. Improvement in performance occurred for the models that integrate the MP statistics under the permanence of ratios assumption.
- A mine planning measure of performance is defined, by calculating the quantity of metal from the validation blasthole data, and comparing the improvement in the estimation of this parameter from the simulated models. Again, the permanence of ratios assumption appeared as the best methodology to integrate MP statistics into the indicator simulation.

6.2 Future Work

Several issues were not addressed in this research and remain as key research topics for integrating multiple-point statistics into geostatistical simulation:

- The assumption of stationarity between different sources of data. As illustrated in the case study (**Chapter 5**), slight differences between the univariate distributions of the data to extract two-point and multiple-point statistics may have large impact on the simulated models. The simple correction developed in **Chapter 5** to make the estimators unbiased showed excellent results. However, this correction can be seen as a re-scaling of all the probabilities of multiple-point events. The consequences of this correction should be further investigated.
- The spacing of the simulated nodes was set to be equal to the spacing of the data used for MP statistical inference. The necessity of a denser simulated grid would require modelling the multiple-point statistics for distances shorter than the spacing between the data used for their inference, in the same way the nugget effect is extrapolated in variogram inference. Positive definiteness of this model is a difficult problem to solve.
- The problem of order relation deviations in sequential indicator simulation and in the updating techniques proposed remains as a drawback of indicator methods. The consequences of constraining the kriging weights to avoid deviations could be of interest.
- Support and precision of different sources of data. The consequences of having data at different support has not been considered in this research, although this could become an issue when implementing the techniques proposed in petroleum applications, where the integration of seismic data is of interest. Seismic data have a very large support compared to well data. For mining applications, the same problem could be foreseen when considering samples with different precision, such as channel samples, blasthole, and drillhole data.

Further extensions of this work would be the application of these techniques to categorical variables. Considering cokriging between thresholds instead of IK probabilities is also an avenue of research.

The determination of methods that control connectivity using only two-point statistics appears as promising. The use of the disjunctive kriging framework, with bivariate isofactorial families that are non-Gaussian could be explored as an alternative method to multiple-point integration.

The relationship between runs and the indicator variogram could be further explored.

An alternative approach suitable for continuous and categorical variables is the use of several classes. The larger the number of classes and points considered at the same time, the larger the possible combinations. This approach is an interesting area for further developments of multiple-point techniques.

Bibliography

[1] F. G. Alabert. The practice of fast conditional simulations through the LU decomposition of the covariance matrix. *Mathematical Geology*, 19(5):369–386, 1987.

[2] A. S. Almeida and A. G. Journel. Joint simulation of multiple variables with a Markov-type coregionalization model. *Mathematical Geology*, 26(5):565–588, 1994.

[3] M. Armstrong. Common problems seen in variograms. *Mathematical Geology*, 16(3):305–313, 1984.

[4] M. Armstrong. Improving the estimation and modeling of the variogram. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for Natural Resources Characterization*, volume 1, pages 1–20. Reidel, Dordrecht, Holland, 1984.

[5] M. Armstrong and R. Jabin. Variogram models must be positive-definite. *Mathematical Geology*, 13(5):455–459, 1981.

[6] M. Armstrong and G. Matheron. New types of disjunctive kriging. Internal note N-943, Centre de Géostatistique, Fontainebleau, 23 pages, 1985.

[7] B. G. Arpat, J. Caers, and A. Haas. Characterization of West-Africa submarine channel reservoirs: A neural network based approach to integration of seismic data. In *2001 SPE Annual Technical Conference and Exhibition*, New Orleans, LA, September 2001. Society of Petroleum Engineers. SPE paper # 71345.

[8] J. Besag. On the statistical analysis of dirty pictures. *J. R. Statistical Society B*, 48(3):259–302, 1986.

[9] G. Bourgault. Statistical declustering and convex estimation using determinant of redundant matrix. In E. Y. Baafi and N. A. Schofield, editors, *Geostatistics Wollongong '96*, volume 1, pages 103–114. Kluwer, 1997.

[10] G. E. P. Box and M. E. Muller. A note on the generation of random normal deviates. *Annals of Mathematical Statistics*, 29(2):610–611, June 1958.

[11] L. Breiman and J. H. Friedman. Estimating optimal transformations for multiple regression and correlation. *Journal of the American Statistical Association*, 80(391):580–598, September 1985.

[12] P. Brooker. Two-dimensional simulations by turning bands. *Mathematical Geology*, 17(1):81–90, 1985.

[13] J. Caers. Stochastic simulation with neural networks. In *Report 11, Stanford Center for Reservoir Forecasting*, Stanford, CA, May 1998.

[14] J. Caers. Markov chain theory for spatial stochastic simulation. In *Report 12, Stanford Center for Reservoir Forecasting*, Stanford, CA, May 1999.

[15] J. Caers and A. G. Journel. Stochastic reservoir simulation using neural networks trained on outcrop data. In *1998 SPE Annual Technical Conference and Exhibition*, pages 321–336, New Orleans, LA, September 1998. Society of Petroleum Engineers. SPE paper # 49026.

[16] J. Caers and X. Ma. Modeling conditional distributions of facies from seismic using neural nets. *Mathematical Geology*, 34(2):143–167, February 2002.

[17] J. Carr and N. Mao. A general form of probability kriging for estimation of the indicator and uniform transforms. *Mathematical Geology*, 25(4):425–438, 1993.

[18] R. Casar-Gonzalez and V. Suro-Perez. Two procedures for stochastic simulation of vuggy formations. In *SPE Latin American and Caribbean Petroleum Engineering Conference*, Buenos Aires, Argentina, March 2001. Society of Petroleum Engineers. SPE paper # 69663.

[19] G. Casella and R. L. Berger. *Statistical Inference.* Duxbury Press, Belmont, California, 1990.

[20] J. P. Chilès and P. Delfiner. *Geostatistics Modeling Spatial Uncertainty.* John Wiley & Sons, New York, 1999.

[21] C. F. Chung. Use of the jacknife method to estimate autocorrelation functions (or variograms). In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for Natural Resources Characterization*, volume 1, pages 55–70. Reidel, Dordrecht, Holland, 1984.

[22] W. G. Cochran. The χ^2 test of goodness of fit. *Annals of Mathematical Statistics*, 23:315–345, September 1952.

[23] N. Cressie. Fitting variogram models by weighted least squares. *Mathematical Geology*, 17(5):563–586, 1985.

[24] N. Cressie. *Statistics for Spatial Data.* John Wiley & Sons, New York, 1991.

[25] N. Cressie and D. M. Hawkins. Robust estimation of the variogram: I. *Mathematical Geology*, 12(2):115–125, 1980.

[26] M. Dagbert. Nested indicator approach for ore reserve estimation in highly variable mineralization. In *92nd Annual General Meeting of CIM - 1990*, Ottawa, Ontario, May 1990.

[27] D. A. Darling. The Kolmogorov-Smirnov, Cramer-von Mises tests. *Annals of Mathematical Statistics*, 28(4):823–838, December 1957.

[28] M. David. *Geostatistical Ore Reserve Estimation.* Elsevier, Amsterdam, 1977.

[29] B. Davis. Indicator kriging as applied to an alluvial gold deposit. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 337–348. Reidel, Dordrecht, Holland, 1984.

[30] M. W. Davis. Generating large stochastic simulation - the matrix polynomial approximation method. *Mathematical Geology*, 19(2):99–107, 1987.

[31] M. W. Davis. Production of conditional simulations via the LU decomposition of the covariance matrix. *Mathematical Geology*, 19(2):91–98, 1987.

[32] P. Delfiner. Linear estimation of non-stationary spatial phenomena. In M. Guarascio, M. David, and C. Huijbregts, editors, *Advanced Geostatistics in the Mining Industry*, pages 49–68, Dordrecht, Holland, 1976. Reidel.

[33] C. V. Deutsch. DECLUS: A Fortran 77 program for determining optimum spatial declustering weights. *Computers & Geosciences*, 15(3):325–332, 1989.

[34] C. V. Deutsch. *Annealing Techniques Applied to Reservoir Modeling and the Integration of Geological and Engineering (Well Test) Data.* PhD thesis, Stanford University, Stanford, CA, 1992.

[35] C. V. Deutsch. A comparative study of pseudo-random number generators. In *Report 5*, Stanford, CA, March 1992. Stanford Center for Reservoir Forecasting.

[36] C. V. Deutsch. Kriging in a finite domain. *Mathematical Geology*, 25(1):41–52, 1993.

[37] C. V. Deutsch. Algorithmically-defined random function models. In R. Dimitrakopoulos, editor, *Geostatistics for the Next Century*, pages 422–435. Kluwer, Dordrecht, Holland, 1994.

[38] C. V. Deutsch. Kriging with strings of data. *Mathematical Geology*, 26(5):623–638, 1994.

[39] C. V. Deutsch. Direct assessment of local accuracy and precision. In E. Y. Baafi and N. A. Schofield, editors, *Geostatistics Wollongong '96*, volume 1, pages 115–125. Kluwer, 1997.

[40] C. V. Deutsch. Cleaning categorical variable (lithofacies) realizations with maximum a-posteriori selection. *Computers & Geosciences*, 24(6):551–562, 1998.

[41] C. V. Deutsch. *Geostatistical Reservoir Modeling.* Oxford University Press, New York, 2002.

[42] C. V. Deutsch and E. Gringarten. Accounting for multiple-point continuity in geostatistical modeling. In *6th International Geostatistics Congress*, Cape Town, South Africa, April 2000. Geostatistical Association of Southern Africa.

[43] C. V. Deutsch and A. G. Journel. *GSLIB: Geostatistical Software Library and User's Guide*. Oxford University Press, New York, 2nd edition, 1998.

[44] C. V. Deutsch and L. Wang. Hierarchical object-based stochastic modeling of fluvial reservoirs. *Mathematical Geology*, 28(7):857–880, 1996.

[45] C. V. Deutsch and X. H. Wen. Integrating large-scale soft data by simulated annealing and probability constraints. *Mathematical Geology*, 32(1):49–68, 2001.

[46] P. A. Dowd. Lognormal kriging - the general case. *Mathematical Geology*, 14(5):475–498, 1982.

[47] P. A. Dowd. The variogram and kriging: Robust and resistant estimates. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 91–106. Reidel, Dordrecht, Holland, 1984.

[48] P. M. Doyen, L. D. den Boer, and W. R. Pillet. Seismic porosity mapping in the Ekofisk field using a new form of collocated cokriging. In *1996 SPE Annual Technical Conference and Exhibition Formation Evaluation and Reservoir Geology*, pages 21–30, Denver, CO, October 1996. Society of Petroleum Engineers. SPE Paper Number 36498.

[49] P. M. Doyen, D. E. Psaila, L. D. D. Boer, and D. Jans. Reconciling data at seismic and well log scales in 3D earth modelling. In *1997 SPE Annual Technical Conference and Exhibition Formation Evaluation and Reservoir Geology*, pages 465–474, San Antonio, TX, October 1997. Society of Petroleum Engineers. SPE paper # 38698.

[50] P. M. Doyen, D. E. Psaila, and S. Strandenes. Bayesian sequential indicator simulation of channel sands from 3D seismic data in the Oseberg field, Norwegian North Sea. In *1996 SPE Annual Technical Conference and Exhibition Formation Evaluation and Reservoir Geology*, pages 197–211, New Orleans, LA, October 1994. Society of Petroleum Engineers. SPE paper # 28382.

[51] M. R. Dunn. A simple sufficient condition for a variogram to yield positive variances under restrictions. *Mathematical Geology*, 15(4):553–564, 1983.

[52] X. Emery. Conditional simulation of non-Gaussian random functions. *Mathematical Geology*, 34(1):79–100, 2002.

[53] X. Emery. *Simulación Estocástica y Geoestadística No Lineal.* To Be Published by Departamento de Ingeniería de Minas - Universidad de Chile, 487 pages, 2003.

[54] C. L. Farmer. Numerical rocks. In P. R. King, editor, *The Mathematical Generation of Reservoir Geology*, Oxford, 1992. Clarendon Press. (Proceedings of a conference held at Robinson College, Cambridge, 1989).

[55] R. A. Fisher and F. Yates. *Statistical Tables for Biological, Agricultural and Medical Research.* Edinburg, 1938.

[56] J. N. Franklin. On the equidistribution of pseudo-random numbers. *Quarterly of Applied Mathematics*, 16:183–188, 1958.

[57] J. C. Fu and M. V. Koutras. Distribution theory of runs: a Markov chain approach. *Journal of the American Statistical Association*, 89(427):1050–1058, September 1994.

[58] L. S. Gandin. *Objective Analysis of Meteorological Fields*. Gidrometeorologicheskoe Izdatel'stvo (GIMEZ), Leningrad, 1963. Reprinted by Israel Program for Scientific Translations, Jerusalem, 1965.

[59] F. Gebhardt. Generating pseudo-random numbers by shuffling a Fibonacci sequence. *Mathematics of Computation*, 21(100):708–709, October 1967.

[60] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, November 1984.

[61] M. G. Genton. Highly robust variogram estimation. *Mathematical Geology*, 30(2):213–221, 1998.

[62] F. Georgsen and H. Omre. Combining fibre processes and Gaussian random functions for modelling fluvial reservoirs. In A. Soares, editor, *Geostatistics Tróia '92*, volume 1, pages 425–440. Kluwer, 1993.

[63] I. M. Glacken. Change of support by direct conditional block simulation. Master's thesis, Stanford University, Stanford, CA, 1996.

[64] A. Goldberger. Best linear unbiased prediction in the generalized linear regression model. *Journal of the American Statistical Association*, 57:369–375, 1962.

[65] J. Gómez-Hernández. Issues on environmental risk assessment. In E. Y. Baafi and N. A. Schofield, editors, *Geostatistics Wollongong '96*, volume 1, pages 15–26. Kluwer, 1997.

[66] J. J. Gómez-Hernández and R. M. Srivastava. ISIM3D: An ANSI-C three dimensional multiple indicator conditional simulation program. *Computers & Geosciences*, 16(4):395–410, 1990.

[67] L. A. Goodman. Simplified runs tests and likelihood ratio tests for Markoff chains. *Biometrika*, 45(1/2):181–197, June 1958.

[68] P. Goovaerts. Comparative performance of indicator algorithms for modeling conditional probability distribution functions. *Mathematical Geology*, 26(3):385–410, 1994.

[69] P. Goovaerts. Stochastic simulation of categorical variables using a classification algorithm and simulated annealing. *Mathematical Geology*, 28(7):909–921, 1996.

[70] P. Goovaerts. Accounting for local uncertainty in environmental decision-making processes. In E. Y. Baafi and N. A. Schofield, editors, *Geostatistics Wollongong '96*, volume 2, pages 929–940. Kluwer, 1997.

[71] P. Goovaerts. *Geostatistics for Natural Resources Evaluation.* Oxford University Press, New York, 1997.

[72] P. Goovaerts. Ordinary cokriging revisited. *Mathematical Geology*, 30(1):21–42, 1998.

[73] B. F. Green, J. E. Keith Smith, and L. Klem. Empirical tests of an additive random number generator. *Journal of the ACM (Association for Computing Machinery)*, 6(4):527–537, October 1959.

[74] M. Greenberger. Notes on a new pseudo-random number generator. *Journal of the ACM (Association for Computing Machinery)*, 8:163–167, 1961.

[75] E. Gringarten and C. V. Deutsch. Teacher's aide - Variogram interpretation and modeling. *Mathematical Geology*, 33(4):507–534, 2001.

[76] F. Guardiano and M. Srivastava. Multivariate geostatistics: Beyond bivariate moments. In A. Soares, editor, *Geostatistics Tróia '92*, volume 1, pages 133–144. Kluwer, 1993.

[77] F. B. Guardiano and R. M. Srivastava. Borrowing complex geometries from training images: The extended normal equations algorithm. In *Report 5*, Stanford, CA, May 1992. Stanford Center for Reservoir Forecasting.

[78] K. Guertin and J. P. Villeneuve. Estimation and mapping of rank related uniform transforms of ion deposition from acid precipitation. In M. Armstrong, editor, *Geostatistics*, volume 2, pages 699–712. Kluwer, 1989.

[79] D. L. Harnett. *Statistical Methods.* Addison-Wesley, third edition, 1982.

[80] J. W. Harris and H. Stocker. *Handbook of Mathematics and Computational Science.* Springer, 1998.

[81] C. J. Huijbregts and G. Matheron. Universal kriging - An optimal approach to trend surface analysis. In *Decision Making in the Mineral Industry*, pages 159–169. Canadian Institute of Mining and Metallurgy, 1971. Special Volume 12.

[82] E. H. Isaaks and R. M. Srivastava. *An Introduction to Applied Geostatistics.* Oxford University Press, New York, 1989.

[83] A. Journel. Geostatistics: Roadblocks and challenges. In A. Soares, editor, *Geostatistics Tróia '92*, volume 1, pages 213–224. Kluwer, 1993.

[84] A. G. Journel. The lognormal approach to predicting local distributions of selective mining unit grades. *Mathematical Geology*, 12(4):285–303, 1980.

[85] A. G. Journel. The indicator approach to estimation of spatial distributions. In *Proceedings of the 17th International APCOM Symposium*, pages 793–806. Society of Mining Engineers, 1982.

[86] A. G. Journel. Nonparametric estimation of spatial distribution. *Mathematical Geology*, 15(3):445–468, 1983.

[87] A. G. Journel. The place of non-parametric geostatistics. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 307–335. Reidel, Dordrecht, Holland, 1984.

[88] A. G. Journel. *Fundamentals of Geostatistics in Five Lessons.* Volume 8 Short Course in Geology. American Geophysical Union, Washington, D. C., 1989.

[89] A. G. Journel. Resampling from stochastic simulations. *Environmental and Ecological Statistics*, 1:63–84, 1994.

[90] A. G. Journel. Combining knowledge from diverse sources: An alternative to traditional data independence hypotheses. *Mathematical Geology*, 34(5):573–596, July 2002.

[91] A. G. Journel and F. Alabert. Non-Gaussian data expansion in the Earth Sciences. *Terra Nova*, 1:123–134, 1989.

[92] A. G. Journel and F. Alabert. New method for reservoir mapping. *J. of Pet. Technology*, pages 212–218, February 1990.

[93] A. G. Journel and C. V. Deutsch. Rank order geostatistics: A proposal for a unique coding and common processing of diverse data. In E. Y. Baafi and N. A. Schofield, editors, *Geostatistics Wollongong '96*, volume 1, pages 174–187. Kluwer, 1997.

[94] A. G. Journel and C. J. Huijbregts. *Mining Geostatistics.* Academic Press, New York, 1978.

[95] A. G. Journel and E. H. Isaaks. Conditional indicator simulation: Application to a Saskatchewan uranium deposit. *Mathematical Geology*, 16(7):685–718, 1984.

[96] A. G. Journel and D. Posa. Characteristic behavior and order relations for indicator variograms. *Mathematical Geology*, 22(8):1011–1025, 1990.

[97] A. G. Journel and M. E. Rossi. When do we need a trend model in kriging? *Mathematical Geology*, 21(7):715–739, 1989.

[98] A. G. Journel and W. Xu. Posterior identification of histograms conditional to local data. *Mathematical Geology*, 26(3):323–359, 1994.

[99] M. G. Kendall and B. B. Smith. Randomness and random sampling numbers. *Journal of the Royal Statistical Society*, 101(1):147–166, 1938.

[100] M. G. Kendall and B. B. Smith. Second paper on random sampling numbers. *Supplement to the Journal of the Royal Statistical Society*, 6(1):51–61, 1939.

[101] W. J. Kennedy Jr. and J. E. Gentle. *Statistical Computing.* Marcel Dekker, Inc., New York, 1980.

[102] A. I. Khuri. *Advanced Calculus with Applications in Statistics.* John Wiley and Sons., Inc., New York, 1993.

[103] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, May 1983.

[104] D. E. Knuth. *The Art of Computer Programming*, volume 2, Seminumerical Algorithms. Addison-Wesley, 1969.

[105] D. G. Krige. A statistical approach to some mine valuations and allied problems at the Witwatersrand. Master's thesis, University of Witwatersrand, South Africa, 1951.

[106] D. H. Lehmer. Mathematical methods in large scale computing units. In *Proceedings of the Second Symposium on Large Scale Digital Computing Machinery*, pages 141–146, Cammbridge, 1951. Harvard University Press.

[107] I. C. Lemmer. Estimating local recoverable reserves via indicator kriging. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 349–364. Reidel, Dordrecht, Holland, 1984.

[108] H. Levene and J. Wolfowitz. The covariance matrix of runs up and down. *Annals of Mathematical Statistics*, 15(1):58–69, March 1944.

[109] D. G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, New York, 1969.

[110] G. R. Luster. *Raw Materials for Portland Cement: Applications of Conditional Simulation of Coregionalization*. PhD thesis, Stanford University, Stanford, CA, 1985.

[111] M. D. MacLaren and G. Marsaglia. Uniform random number generators. *Journal of the ACM (Association for Computing Machinery)*, 12(1):83–89, January 1965.

[112] A. Marechal. Kriging seismic data in presence of faults. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 271–294. Reidel, Dordrecht, Holland, 1984.

[113] G. Marsaglia. A convenient method for generating normal variables. *SIAM Review*, 6(3):260–264, July 1964.

[114] G. Marsaglia. The structure of linear congruential sequences. In S. K. Zaremba, editor, *Applications of Number Theory to Numerical Analysis*, pages 249–285. Academic Press, London, 1972.

[115] B. Matérn. *Spatial Variation*, volume 36 of *Lecture Notes in Statistics*. Springer Verlag, New York, second edition, 1980. First edition published by Meddelanden fran Statens Skogsforskningsinstitut, Band 49, No. 5, 1960.

[116] G. Matheron. *Traité de Géostatistique Appliquée*. ed. Technip, Paris, 1962. Vol. 1 (1962), Vol. 2 (1963).

[117] G. Matheron. *Les variables régionalisées et leur estimation*. Masson et Cie. Editeurs, Paris, 1965.

[118] G. Matheron. Le krigeage disjonctif. Internal note N-360, Centre de Géostatistique, Fontainebleau, 40 pages, 1973.

[119] G. Matheron. A simple substitute for conditional expectation: the disjunctive kriging. In M. Guarascio, M. David, and C. Huijbregts, editors, *Advanced Geostatistics in the Mining Industry*, pages 221–236, Dordrecht, Holland, 1976. Reidel.

[120] G. Matheron. La déstructuration des hautes teneurs et le krigeage des indicatrices. Internal note N-761, Centre de Géostatistique, Fontainebleau, 1982.

[121] G. Matheron. Isofactorial models and change of support. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 449–467. Reidel, Dordrecht, Holland, 1984.

[122] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21(6):1087–1092, June 1953.

[123] A. M. Mood. The distribution theory of runs. *Annals of Mathematical Statistics*, 11(4):367–392, December 1940.

[124] F. Mosteller. Note on an application of runs to quality control charts. *Annals of Mathematical Statistics*, 12(2):228–232, June 1941.

[125] D. E. Myers. Cokriging - new developments. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 295–305. Reidel, Dordrecht, Holland, 1984.

[126] D. E. Myers. Pseudo-cross variograms, positive-definiteness, and cokriging. *Mathematical Geology*, 23(6):805–816, 1991.

[127] J. V. Neumann. Various techniques used in connection with random digits. In *National Bureau of Standards symposium, NBS Applied Mathematics Series 12*, Washington, D. C., 1951. National Bureau of Standards.

[128] R. A. Olea, editor. *Geostatistical Glossary and Multilingual Dictionary*. Oxford University Press, New York, 1991.

[129] R. A. Olea. Fundamentals of semivariogram estimation, modeling, and usage. In J. M. Yarus and R. L. Chambers, editors, *Stochastic Modeling and Geostatistics: Principles, Methods, and Case Studies*, pages 27–36. AAPG Computer Applications in Geology, No. 3, 1995.

[130] H. Omre. The variogram and its estimation. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 107–125. Reidel, Dordrecht, Holland, 1984.

[131] J. Ortiz C. Characterization of high order correlation for enhanced indicator simulation. In *Centre For Computational Geostatistics*, volume 3, Edmonton, AB, 2001.

[132] J. Ortiz C. Characterization of high order correlation for enhanced indicator simulation. Ph. D. Dissertation Proposal, March 2001.

[133] J. Ortiz C. Research note: HISIM - Hierarchical Indicator Simulation. In *Centre For Computational Geostatistics*, volume 3, Edmonton, AB, 2001.

[134] J. Ortiz C. and C. V. Deutsch. Calculation of uncertainty in the variogram. *Mathematical Geology*, 34(2):169–183, 2002.

[135] J. Ortiz C. and C. V. Deutsch. Hierarchical indicator simulation. In *Proceedings of the 30th International APCOM Symposium*, Phoenix,AZ, February 2002. Society of Mining Engineers.

[136] B. Oz and C. V. Deutsch. Size scaling of cross-correlation between multiple variables. In *Centre For Computational Geostatistics*, volume 3, Edmonton, AB, 2001.

[137] H. Parker, A. G. Journel, and W. Dixon. The use of conditional lognormal probability distributions for the estimation of open pit ore reserves in stratabound uranium deposits: A case study. In *Proceedings of the 16th International APCOM Symposium*, pages 133–148, Tucson, AZ, October 1979. Society of Mining Engineers.

[138] M. J. Pyrcz and C. V. Deutsch. Debiasing for improved inference of the one point statistic. In *Proceedings of the 30th International APCOM Symposium*, Phoenix,AZ, February 2002. Society of Mining Engineers.

[139] W. Y. Qiu and M. G. Kelkar. Simulation of geological models using multipoint histogram. In *1995 SPE Annual Technical Conference and Exhibition*, Dallas, TX, October 1995. Society of Petroleum Engineers. SPE paper # 30601.

[140] RAND Corporation. *A Million Random Digits with 100,000 Normal Deviates.* Free Press, Glencoe, IL, 1955.

[141] J. Rendu. Disjunctive kriging: Comparison of theory with actual results. *Mathematical Geology*, 12(4):305–320, 1980.

[142] J. M. Rendu. *An Introduction to Geostatistical Methods of Mineral Evaluation.* South African Institute of Mining and Metallurgy, Johannesburg, 1978.

[143] J. M. Rendu. Normal and lognormal estimation. *Mathematical Geology*, 11(4):407–422, 1979.

[144] B. D. Ripley. *Stochastic Simulation.* John Wiley & Sons, New York, 1987.

[145] J. Rivoirard. *Introduction to Disjunctive Kriging and Non-Linear Geostatistics.* Oxford University Press, 181 pages, New York, 1994.

[146] A. Rotenberg. A new pseudo-random number generator. *Journal of the ACM (Association for Computing Machinery)*, 7(1):75–77, January 1960.

[147] D. H. Rothman. Nonlinear inversion, statistical mechanics, and residual statics estimation. *Geophysics*, 50:2784–2796, 1985.

[148] L. Sandjivy. The factorial kriging analysis of regionalized data. its application to geochemical prospecting. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 559–571. Reidel, Dordrecht, Holland, 1984.

[149] P. W. Shaughnessy. Multiple runs distributions: Recurrences and critical values. *Journal of the American Statistical Association*, 76(375):732–736, September 1981.

[150] H. S. Sichel. New methods in the statistical evaluation of mine sampling data. *Transactions of the Institution for Mining and Metallurgy*, March 1951-1952.

[151] A. Soares. Short note: Sequential indicator simulation with correction for local probabilities. *Mathematical Geology*, 30(6):761–765, 1998.

[152] A. R. Solow. Mapping by simple indicator kriging. *Mathematical Geology*, 18(3):335–352, 1986.

[153] S. Srinivasan and J. Caers. Conditioning reservoir models to dynamic data - a forward modeling perspective. In *2000 SPE Annual Technical Conference and Exhibition*, Dallas, TX, October 2000. Society of Petroleum Engineers. SPE paper # 62941.

[154] R. M. Srivastava. Iterative methods for spatial simulation. In *SCRF report*, Stanford, CA, May 1992.

[155] R. M. Srivastava. An annealing procedure for honouring change of support statistics in conditional simulation. In R. Dimitrakopoulos, editor, *Geostatistics for the Next Century*, pages 277–290. Kluwer, Dordrecht, Holland, 1994.

[156] T. H. Starks and J. H. Fang. The effect of drift on the experimental semivariogram. *Mathematical Geology*, 14(4):309–319, 1982.

[157] S. Strebelle. Sequential simulation drawing structures from training images. In *SCRF report 12*, Stanford, CA, May 1999.

[158] S. Strebelle. Sequential simulation drawing structures from training images. In *SCRF report 13*, Stanford, CA, May 2000.

[159] S. Strebelle and A. G. Journel. Sequential simulation drawing structures from training images. In *6th International Geostatistics Congress*, Cape Town, South Africa, April 2000. Geostatistical Association of Southern Africa.

[160] S. Strebelle and A. G. Journel. Reservoir modeling using multiple-point statistics. In *2001 SPE Annual Technical Conference and Exhibition*, New Orleans, LA, September 2001. Society of Petroleum Engineers. SPE paper # 71324.

[161] B. V. Sukhatme. On certain probability distributions arising from points on a line. *Journal of the Royal Statistical Society, Series B (Methodological)*, 13(2):219–232, 1951.

[162] J. Sullivan. Conditional recovery estimation through probability kriging: theory and practice. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 365–384. Reidel, Dordrecht, Holland, 1984.

[163] P. Switzer and H. M. Parker. The problem of ore versus waste discrimination for individual blocks: The lognormal model. In M. Guarascio, M. David, and C. Huijbregts, editors, *Advanced Geostatistics in the Mining Industry*, pages 203–218, Dordrecht, Holland, 1976. Reidel.

[164] W. E. Thomson. A modified congruence method of generating pseudo-random numbers. *Computer Journal*, 1(2):83,86, July 1958.

[165] L. H. C. Tippett. Random sampling numbers. *Tracts for Computers*, XV, 1927.

[166] T. T. Tran. Improving variogram reproduction on dense simulation grids. *Computers & Geosciences*, 20(7):1161–1168, 1994.

[167] G. Verly. The multiGaussian approach and its applications to the estimation of local reserves. *Mathematical Geology*, 15(2):259–286, 1983.

[168] G. Verly. The block distribution given a point multivariate normal distribution. In G. Verly, M. David, A. G. Journel, and A. Marechal, editors, *Geostatistics for natural resources characterization*, volume 1, pages 495–515. Reidel, Dordrecht, Holland, 1984.

[169] G. Verly and J. Sullivan. MultiGaussian and probability krigings - application to the Jerritt Canyon deposit. *Mining Engineering*, pages 568–574, June 1985.

[170] S. Viseur. Stochastic boolean simulation of fluvial deposits: A new approach combining accuracy with efficiency. In *1999 Annual Technical Conference and Exhibition*. Society of Petroleum Engineers, October 1999. SPE Paper Number 56688.

[171] H. Wackernagel. Cokriging versus kriging in regionalized multivariate data analysis. *Geoderma*, 62:83–92, 1994.

[172] H. Wackernagel, P. Petitgas, and Y. Touffait. Overview of methods for coregionalization analysis. In M. Armstrong, editor, *Geostatistics*, volume 1, pages 409–420. Kluwer, 1989.

[173] L. Wang. Modeling complex reservoir geometries with multiple-point statistics. Technical report, Stanford Center for Reservoir Forecasting, Stanford, CA, May 1995.

[174] W. J. Westlake. A uniform random number generator based on the combination of two congruential generators. *Journal of the ACM (Association for Computing Machinery)*, 14(2):337–340, April 1967.

[175] R. S. Wikramaratna. ACORN - a new method for generating sequences of uniformly distributed pseudo-random numbers. *Journal of Computational Physics*, 83:16–31, 1989.

[176] R. S. Wikramaratna. ACORN random number generator user documentation. User Documentation, October 1990.

[177] R. S. Wikramaratna. Theoretical analysis of the ACORN random number generator, 1990. SIAM Conference on Applied Probability in Science and Engineering.

[178] W. Xu and A. G. Journel. Conditional curvilinear stochastic simulation using pixel-based algorithms. In *SCRF report*, Stanford, CA, May 1995.

[179] W. Xu, T. T. Tran, R. M. Srivastava, and A. G. Journel. Integrating seismic data in reservoir modeling: the collocated cokriging alternative. In *67th Annual Technical Conference and Exhibition*, pages 833–842, Washington, DC, October 1992. Society of Petroleum Engineers. SPE paper # 24742.

[180] H. Zhu and A. Journel. Formatting and integrating soft data: Stochastic imaging via the Markov-Bayes algorithm. In A. Soares, editor, *Geostatistics Tróia 1992*, volume 1, pages 1–12. Kluwer, 1993.

Appendix A

Pseudo-Random Number Generators

Random numbers are at the heart of all geostatistical simulation methods. Practitioners assume that the software they are using uses an appropriate pseudo-random number generator; however, this may not be true. This Appendix contains a short literature review on random number generators and tests to quantify their randomness. The history of pseudo-random number generation is reviewed. Tests for randomness are described and applied to five different pseudo random number generators. Results show that generators that were considered good a few years ago fail some recent tests. We recommend careful testing and monitoring of the literature.

A.1 Random Number Generators

In the early twentieth century people needed random numbers for their scientific work. They started replacing the basic methods of drawing balls of a well stirred urn or rolling dice with tables of numbers taken from some source or with random numbers generated by mechanical devices. In 1927 a table of over 40,000 digits taken at random from census reports was published by L. H. C. Tippett [165]. M. G. Kendall and B. Babington-Smith [99] presented in 1938 a mechanical device to generate random digits. They proposed 4 different tests that they applied to a sequence of 5,000 digits generated by their machine. The same tests were applied to two series of 1,000 digits obtained from Tippett's table. All the sequences passed the tests and were considered locally random. In 1939 a table with 100,000 digits was published by Kendall and Babington-Smith [100] using the same randomizing machine. They tested their digits and those published by Fisher and Yates [55] with satisfactory results. The well-known RAND [140] table of random digits was published in 1955. It included 1 million digits generated by another machine from electronic noise. Most of those tables showed undesirable properties when new tests were applied.

The introduction of computers led to other ways to generate random number sequences. Tables had limited utility because of their size. Instead, arithmetic operations were proposed to efficiently generate sequences of random numbers on computers. J. Von Neumann [127] presented in 1946 the middle square method, which consists in taking the middle digits of the previous number squared. This method does not generate random sequences [104].

One of the most popular methods to generate sequences of random numbers was the Linear Congruential Method, which is covered in more detail in the following section. These generators present some undesirable properties such as lattice structure [114]. In 1965, M.D. MacLaren and G. Marsaglia presented a procedure to combine two generators to have better sequences (more random). In 1989, R.S. Wikramaratna proposed the Additive Congruential Method which has proved to give satisfactory results. More details about the history of pseudo- random number generators can be found in Knuth [104], Ripley [144], and Kennedy and Gentle [101].

A.1.1 Linear Congruential Method

D. H. Lehmer [106] introduced in 1948 the idea of generating a random number sequence using the following formula:

$$X_{n+1} = (aX_n + c)_{modM}, \quad n > 1$$

where X_0 is the starting value or seed of the sequence ($X_0 \geq 0$), a is called the multiplier ($a \geq 0$), c is the increment ($c \geq 0$), and M is the modulus ($M \geq X_0$, $M \geq a$, $M \geq c$).

When c is set to zero (as it was in the original sequence proposed by Lehmer) the method is called *Multiplicative Congruential Method*, otherwise (i.e. if $c \neq 0$), it is called *Mixed Congruential Method.* The first examples of the mixed generator were given independently by Thomson [164] and Rotenberg [146]. Other applications were presented by Franklin [56] and Greenberger [74].

A.1.2 Additive Congruential Method

Additive generators calculate each number as some additive combination of the previous n numbers in the sequence. R. S. Wikramaratna [175, 176, 177] proposed the k^{th} order ACORN (additive congruential random number) generator X_j^k, a more general recursive method than the linear congruential, which combines the previous number in the sequence with a corresponding number from the $(k-1)^{th}$ order sequence. X_j^k is defined recursively from a seed X_0^0 ($0 < X_0^0 < 1$) and a set of k initial values X_0^m, m=1, ..., k each satisfying $0 \leq X_0^m \leq 1$ by:

$$\begin{array}{ll} X_n^0 = X_{n-1}^0, & n \geq 1 \\ X_n^m = (X_n^{m-1} + X_{n-1}^m)_{mod1}, & n \geq 1, \quad m = 1, ..., k \end{array}$$

This generator has three features: it is faster to compute (the algorithm is very simple), the period length can be set arbitrarily large, and it gives the same sequence in any machine (differing only in the number of significant digits).

Figure A.1 presents a schematic of this method. The user has to choose the numbers in the first column. All the numbers in the Zero Order row are the same (the seed number X_0^0). The arrows show which previous numbers are used to calculate the current one. The numbers generated in the row of the k^{th} order are considered to be pseudo-random numbers. One should not take the first few numbers, since for seed numbers close to each other they may be similar. It is recommended to initialize the sequence not considering the first thousands.

$$\begin{tikzcd}
Zero\ Order: & X_0^0 \arrow[r] & X_1^0 \arrow[r] \arrow[d] & X_2^0 \arrow[r] \arrow[d] & \cdots \arrow[r] & X_N^0 \arrow[d] \\
1^{st}\ Order: & X_0^1 \arrow[r] & X_1^1 \arrow[r] \arrow[d] & X_2^1 \arrow[r] \arrow[d] & \cdots \arrow[r] & X_N^1 \arrow[d] \\
2^{nd}\ Order: & X_0^2 \arrow[r] & X_1^2 \arrow[r] \arrow[d] & X_2^2 \arrow[r] \arrow[d] & \cdots \arrow[r] & X_N^2 \arrow[d] \\
 & & \vdots \arrow[d] & \vdots \arrow[d] & & \vdots \arrow[d] \\
k^{th}\ Order: & X_0^k \arrow[r] & X_1^k \arrow[r] & X_2^k \arrow[r] & \cdots \arrow[r] & X_N^k
\end{tikzcd}$$

Figure A.1: Schematic showing how **acorn** generates random numbers.

A.1.3 Other Methods

Many other methods to generate sequences of random numbers may be cited. R. R. Coveyou created a quadratic method (which is, in fact, a double precision middle square method). The seed has to be chosen such that:

$$X_{0 mod4} = 2$$

and the sequence is then defined by:

$$X_{n+1} = X_n \cdot (X_n + 1)_{mod2^e}, \quad n > 0$$

The well-known Fibonacci sequence (originated in the early 1950's) present a long period (longer than M), but it is not satisfactorily random. It is defined by:

$$X_{n+1} = (X_n + X_{n-1})_{modM}$$

Variations such as the one presented by Green, Smith and Klem [73], do not improve the randomness of the sequences considerably:

$$X_{n+1} = (X_n + X_{n-k})_{modM}$$

An interesting approach was presented by MacLaren and Marsaglia [111] and consists in combining two sequences to get another "even more random". This approach has been accepted by some authors and rejected by others [35, 144]. In any case, the algorithm proposed by MacLaren and Marsaglia seems to work well, as shown by F. Gebhardt [59].

A different method to combine two sequences was proposed by W. J. Westlake [174], based on circular shifting and exclusive "or" on a binary computer.

A.2 Statistical Tests

The sequences generated by any algorithm must be tested in order to know quantitatively its randomness. The statistical tests applied to pseudo- random number sequences can be grouped as:

Empirical tests : Non-parametric test of a sample sequence of numbers. The evaluation is based on "goodness of fit" of observed distributions with respect to expected ones (predicted theoretically).

Theoretical tests : Based on theoretical properties of the generators that may be deduced without a sample sequence. Generators such as the congruential ones are predictable in the sense that knowing the values a, c, M, and X_0, the period of the generator may be predicted, as well as other properties.

A.2.1 Empirical Tests

Many different tests have been used to test sequences of pseudo-random numbers. Most of them are based in a comparison between observed and expected frequencies. A χ^2 test or a Kolmogorov-Smirnov test can be applied to quantify the mismatch between both distributions, based on probability at some level of significance [22, 27]. A brief description of those tests is given below:

1. **χ^2 Test**: The following statistic is used:

   $$Q = \sum_{i=1}^{n} \frac{(x_i - m_i)^2}{m_i} \sim \chi^2_{n-1}$$

   where x_i is the experimental frequency in interval i and m_i is the expected (theoretical) frequency in the same interval. The quadratic form Q follows a χ^2 distribution with $n-1$ degrees of freedom [22]. Once the value of Q has been calculated, the user can refer to tables to find the percentile for a χ^2 distribution with $n-1$ degrees of freedom. One should expect to be between the fifth and ninety fifth percentiles.

2. **Kolmogorov-Smirnov Test**: The statistic D may be compared with the critical value for a given significancy level:

   $$D = Max|F_i - S_i|$$

   where F_i denotes the cumulative relative frequency for each category of the theoretical distribution and S_i is the value from the observed data.

   Tables of critical values of D for different probability values may be found in most statistics books [79, 104].

The following are considered the most powerful tests for randomness [35, 99, 104].

- **Frequency test (uniformity or equidistribution test)**: in a sequence of random digits the observed frequency can be compared with the expected frequency (each digit should appear $\frac{n}{10}$ times, where n is the total number of digits in the series). A χ^2 test can be applied to quantify the departure between observed and expected results. In a sequence of numbers, the interval can be divided into n subsets (e.g. $U \leq 0.01, 0.01 < U \leq 0.02, ...$), the observed and expected frequencies are calculated and a χ^2 test is applied. Srivastava proposed to check uniformity over extreme intervals close to 0 and close to 1 (for applications of random numbers in mining simulations, where extreme values will be critical when transformed to actual grades).

- **Serial test (k-dimensional uniformity)**: Given that numbers in the sequence should be independent, a good random number generator should produce pairs of numbers that uniformly fill the unit square, triplets that uniformly fill the unit cube, etc. The frequency of k-tuples $\{U_i, U_{i+1}, ..., U_{i+k-1}\}$ is calculated and compared with the expected frequency. We expect to have the same number of observations in each one of the n^k equal-sized cells, where n is the number of cells in which each dimension is divided and k is the dimension or size of the k-tuple.

- **Poker test (partition test)**: This test was originally proposed by Kendall and Babington-Smith [99] for sequences of digits. When digits are arranged in blocks of five, there will be certain expectation of the numbers in which the five digits are all the same, the numbers in which there are four of one kind, and so on. Knuth [104] explains the "classical" poker test in the following manner: The numbers are arranged in n groups of five successive integers, $(Y_{5j}, Y_{5j+1}, ..., Y_{5j+4})$, $0 \leq j < n$. We observe which of the following seven patterns each quintuple matches:

All different:	abcde	Full house:	aaabb
One pair:	aabcd	Four of a kind:	aaaab
Two pairs:	aabbc	Five of a kind:	aaaaa
Three of a kind:	aaabc		

  A χ^2 test is based on the number of quintuples in each category. A simpler version is proposed by Knuth. The test can also be applied using only four digits [100].

- **Gap test (Runs above and below the median)**: We can consider the gaps occurring between the same digit in the series. For example, one digit will be followed immediately by the same digit in about one-tenth of the cases (in this case, there will be no gap). There will be one digit (different) between two equal digits in about nine- hundredths of the cases. In about eighty-one-thousandths of the cases there will be a gap of two between a repeated digit, and so on.

  A generalization for a series of numbers (not just digits) would be to examine the length of gaps between occurrences of U_j in a certain range. Let $0 \leq \alpha < \beta \leq 1$, we consider the length of consecutive subsequences $U_j, U_{j+1}, ..., U_{j+r-1}$ in which U_{j-1} and U_{j+r} lies between α and β but the elements in the subsequence do not. This subsequence represents a gap of length r.

  The special cases $(\alpha, \beta) = (0, \frac{1}{2})$ or $(\frac{1}{2}, 1)$ originated the so called tests of runs above and below the mean (or the median). In order to implement this test, we have to produce another sequence from the sequence being tested, by counting the length of successive runs above and/or below the median. For example, the sequence:

$$0.35, 0.56, 0.12, 0.11, 0.84, 0.76, 0.77, 0.45, 0.61, 0.51, ...$$

would generate the following sequence of above/below the median observations:

below, above, below, below, above, above, above, below, above, above, ...

and the sequence of lengths of runs above/below the median would be:

1, 1, 2, 3, 1, 2,...

The same procedure can be applied for any threshold (not only the median). Depending on the proportion of values above and below the threshold, the total number of runs above and below the threshold should follow a normal distribution with the following mean and variance [123]:

$$E[r] = 2 \cdot n \cdot p_A \cdot p_B$$

$$\sigma_r^2 = 4 \cdot n \cdot p_A \cdot p_B \cdot (1 - 3 \cdot p_A \cdot p_B)$$

where p_A and p_B represent the proportion (probability) of values above and below some threshold respectively, and n is the total number of values in the sequence.

When the threshold is the median (or the mean) of a uniform distribution then, the parameters are simply:

$$E[r] = \frac{n}{2} \qquad \sigma_r^2 = \frac{n}{4}$$

A χ^2 test can be applied to the observed values in order to determine if there is any significant difference with the expected value.

- **Runs up and down**: This test examines the length of monotone subsequences of the original sequence, i.e. segments which are increasing or decreasing.

  Again, to implement this test, we have to produce another sequence from the sequence being tested, by counting the length of successive runs up and down. For example, the sequence:

  0.35, 0.56, 0.12, 0.11, 0.84, 0.76, 0.77, 0.45, 0.61, 0.51, ...

  would generate the following sequence of runs up and down:

  up, down, down, up, down, up, down, up, down, ...

  and the sequence of lengths of runs up and down would be:

  1, 2, 1, 1, 1, 1, 1, 1, ...

For an independent and uniform sequence of numbers, the number of runs up and down should come from a normal distribution with the following mean and variance:

$$E[r] = \frac{2 \cdot n - 1}{3}$$

$$\sigma_r^2 = \sqrt{\frac{16 \cdot n - 29}{90}}$$

In this case, the conventional χ^2 test has to be modified to take into account the fact that the number of runs of various lengths are negatively correlated [101, 104, 108] (covariances are used to make this correction).

- **Extreme values (maximum of k)**: Given that in most applications extreme values are important (e.g. in mining simulations), a test over the extreme values of the sequence is required. If we group the sequence into k-tuples, and we extract, for each k-tuple, the maximum value, then the distribution of maximums from k-tuples should show no serial correlation and should have a cumulative distribution that follows a power law: $F_k(x) = x^k$. Now, we must show that the distribution of maximums follows a power law. The probability of $max(U_1, U_2, ..., U_k) \leq x$ is the probability that $U_1 \leq x$ and $U_2 \leq x$ and ... and $U_k \leq x$, and this is the product of the individual probabilities, $x \cdot x \cdot ... \cdot x = x^k$. The closeness of the observed distribution to the expected can be checked comparing the distribution of $(max(U_1, U_2, ..., U_k))^k$ with the uniform distribution, using a χ^2 test. The serial correlation can be calculated. This test can also be applied with the minimum of k.

- **Coupon collector's test**: This test consists in calculate the length of segments required to get at least one observation per cell, when the interval [0,1] is divided in some equally sized number of classes d. A χ^2 test can be applied to the observed counting of length r. The corresponding probabilities are:

$$p_r = \tfrac{d!}{d^r} \left\{ \begin{matrix} r-1 \\ d-1 \end{matrix} \right\}, \quad d \leq r < t \qquad p_t = 1 - \tfrac{d!}{d^{t-1}} \left\{ \begin{matrix} t-1 \\ d \end{matrix} \right\}$$

  where r is the length of the segment, d is the number of classes and t is some length such that all the segments longer than t are put together.

- **Permutation test**: If we divide the sequence in k-tuples, then each k-tuple can have $t!$ possible relative orderings. The number of times each ordering appears is counted and a χ^2 test is applied with $k = t!$ classes and with probability $1/t!$ for each ordering.

- **Test on subsequences**: All the tests previously presented can be applied to a subset of the sequence, so we can verify if these subsets behave equally random than the whole sequence.

A.2.2 Theoretical Tests

Some random number generators are suitable for analysis *a priori*, so that the parameters needed to generate a sequence can be understood and chosen properly. Linear congruential generators have been thoroughly studied [74, 101, 104, 114, 144]. Some other generators, as the additive method presented by Green, Smith and Klem [73] allows some theoretical analysis as well. Wikramaratna [175] shows some theoretical results for his additive congruential generator. The interested reader can check those references for further explanations of the tests.

Some authors recommend against methods that do not allow those analysis [144], such as the one proposed by MacLaren and Marsaglia [111];however those generators have shown to perform well for many applications [35].

A.3 Testing Five Random Number Generators

This section contains the results of testing five different random number generators. The tests presented here are only those which have proven to be the more effective to detect poor pseudo-random number generators [104].

The following methods were tested:

- Linear Congruential Method (`lcorn`): the parameters of this generator are: $m = 2^{16} + 1$, $a = 75$, $c = 1$. Three seed numbers were used: 69069, 112063, and 76715.
- Mixed Congruential Method (`mcorn`): this is the generator proposed by MacLaren and Marsaglia [111].The seeds used are the same than those for `lcorn`.
- Additive Congruential Method (`acorn`): This is the generator proposed by Wikramaratna [175] in real arithmetic. The seeds used must be real values between 0 and 1. In this application the initial values are: 0.10, 0.81, and 0.12.
- Additive Congruential Method (`acorni`): This is the generator proposed by Wikramaratna [177] in integer arithmetic. Again, the seeds used are those for `lcorn`.
- `excel`: this generator comes with the commercial software `Microsoft Excel`. The pseudo-random sequences are generated without specifying a seed number.

For each generator, nine sequences of numbers have been created. Three sequences of 10,000 values, three of 30,000, and three of 90,000 values between 0 and 1.

A.3.1 Serial Correlation Test

The serial correlation was calculated using the routine `gam` of the public domain software GSLIB [39]. Results are presented in **Table A.1**. Correlations greater than 0.02 in absolute value were highlighted. `lcorn` presents four of those high values for sequences of 10,000 numbers, however they are still not significant. `excel` also has one sequence with correlation greater than 0.02. All the random number generators passed this test, since all the correlations are acceptably close to zero.

A.3.2 Uniformity Test

The uniformity of the sequences was tested dividing the interval [0,1] into 100 subintervals ([0,0.01), [0.01,0.02), ..., [0.99,1]). A χ^2 test was applied to the observed

10000 Data						
Algorithm	Seed	h = 1	h = 2	h = 3	h = 4	h = 5
LCORN	69069	0.01241	0.00501	-0.01182	**0.02127**	0.00371
	112063	0.01217	-0.00515	0.00106	0.00048	**-0.02291**
	76715	**0.02587**	0.00015	-0.00788	0.00931	**0.02054**
MCORN	69069	0.00415	0.00214	0.01294	-0.00814	0.01826
	112063	0.01501	-0.01339	0.00885	0.00631	-0.00444
	76715	-0.01881	-0.00113	0.00805	0.00106	0.00052
ACORN	0.10	-0.00323	0.00461	-0.00661	0.00579	0.00957
	0.81	0.01117	-0.00783	0.00027	-0.00634	0.00761
	0.12	0.00106	0.01729	0.00602	-0.00205	-0.00486
ACORNI	69069	-0.00318	-0.00831	-0.00647	0.00002	0.01980
	112063	0.01048	0.00654	0.00904	0.00217	0.01222
	76715	-0.00810	-0.00755	0.00938	-0.01783	0.00373
EXCEL	—	0.00140	-0.00602	0.00419	-0.00220	-0.01285
	—	0.00053	-0.01690	-0.00025	-0.00903	**0.02864**
	—	-0.01667	0.00894	0.01240	-0.00189	0.00137
30000 Data						
Algorithm	Seed	h = 1	h = 2	h = 3	h = 4	h = 5
LCORN	69069	0.01407	-0.00198	-0.00580	0.00700	-0.00875
	112063	0.01490	-0.00093	0.00497	-0.00486	-0.00137
	76715	0.01271	-0.00278	-0.00719	0.00513	-0.00495
MCORN	69069	0.00287	0.00656	0.00737	0.00176	0.01153
	112063	0.00381	-0.00892	0.00192	0.00185	0.00234
	76715	0.00750	-0.00119	0.01183	0.01169	-0.00305
ACORN	0.10	-0.01548	-0.00196	0.00473	0.00011	0.00683
	0.81	-0.00086	-0.00071	0.00155	-0.00761	-0.00681
	0.12	-0.00124	0.01308	-0.00241	0.00494	0.00354
ACORNI	69069	0.00059	-0.01027	-0.00921	0.00255	0.01482
	112063	0.00676	0.00775	0.01339	-0.00249	0.00221
	76715	0.00112	-0.00593	0.00509	-0.01035	-0.00144
EXCEL	—	0.00670	0.00648	-0.00099	-0.00416	0.00308
	—	-0.00762	0.00476	0.00342	-0.00472	0.00165
	—	0.00140	0.00038	-0.00897	0.00318	-0.00352
90000 Data						
Algorithm	Seed	h = 1	h = 2	h = 3	h = 4	h = 5
LCORN	69069	0.01243	0.00017	-0.00173	0.00183	-0.00309
	112063	0.01292	0.00139	0.00140	-0.00179	-0.00167
	76715	0.01454	-0.00013	-0.00205	0.00116	-0.00167
MCORN	69069	0.00345	0.00552	0.00289	-0.00112	0.00471
	112063	0.00154	-0.00329	-0.00122	0.00077	0.00068
	76715	0.00297	-0.00136	0.00255	0.00958	0.00109
ACORN	0.10	-0.00726	0.00494	-0.00331	-0.00377	0.00019
	0.81	0.00509	-0.00195	-0.00065	-0.00388	-0.00622
	0.12	0.00076	0.00376	0.00073	0.00437	0.00008
ACORNI	69069	-0.00177	-0.00184	-0.00487	-0.00057	0.00174
	112063	0.00354	0.00328	0.00482	-0.00270	0.00163
	76715	0.00171	-0.00427	0.00036	-0.00178	0.00024
EXCEL	—	-0.00062	-0.00570	0.00039	0.00339	0.00538
	—	-0.00721	0.00331	-0.00560	0.00153	0.00066
	—	-0.00487	0.00319	0.00007	0.00010	0.00163

Table A.1: Results of serial correlation test for 5 pseudo-random number generators and sequences of length 10000, 30000 and 90000.

frequencies. The results are presented in **Table A.2**. All the χ^2 percentiles outside of the 90% central confidence interval were highlighted. `lcorn` failed this test when sequences of 30,000 and 90,000 numbers were used. `mcorn` and `acorn` seem to perform the best in this test.

A.3.3 K-Dimensional Uniformity Test

After partitioning the space of 3, 4, and 5 dimensions regularly, the frequency of observed values in each subset should be approximately the same if the numbers are random. In 3-D the space was divided into $15^3 = 3375$ cells, in 4-D it was divided into $8^4 = 4096$ cells, and in 5-D it was divided into $5^5 = 3125$ cells. **Table A.3** presents the χ^2 percentiles for this test. `lcorn` failed this test for all the sequences tested. Again, `mcorn` seems to perform the best. `excel` also gives good results. `acorn` and `acorni` give acceptable results.

A.3.4 Runs Up and Down

The total number of runs up and down and the number of runs for each length were calculated and are shown in **Table A.4**. The total number of runs should fall between the 5^{th} and 95^{th} percentile of the expected distribution. For 10,000 numbers, the acceptable interval is (6598,6736), for 30,000 it is (19880,20120), and for 90,000 the confidence interval is (59793,60207).

`lcorn` gives too few runs in most of the sequences, while `mcorn` failed in one sequence, which is acceptable. `acorn` and `acorni` failed in two cases. `excel` performed excellent in this test. In summary, we can say that `lcorn` failed and `excel` gave the best results, and the other generators gave acceptable results.

A.3.5 Runs Above and Below the Median

Table A.5 presents the total number of runs above and below the mean, and the detailed list of number of runs of different lengths. According to the limit distribution of the total number of runs, the observed number of runs should be into the interval $[m - 1.645 \cdot \sigma, m + 1.645 \cdot \sigma]$. That means that for the sequences of 10,000 values, they should be within (4918,5082). In the case of 30,000 numbers, the number of runs should be into (14858,15142),and finally, for 90,000 numbers, it should be in (44753,45247).

The results again show that `lcorn` gives bad results for most of the sequences. The other generators only have minor problems with this test.

A.3.6 Extreme Values

Generators were tested for maximum values in a k-tuple. The distribution of maximums should follow a power law. A χ^2 test was applied to compare the observed frequencies with the expected ones. **Table A.6** presents the percentile of the χ^2 test for each sequence.

`lcorn` failed the test for uniformity ($k = 1$) (see **Table A.1**) and for higher values of k. All the other generator presented some problems, however in general they seemed to pass this test. `mcorn` gave the best results.

10000 Data				
Algorithm	Seed	0.0 - 1.0	0.0 - 0.1	0.9 - 1.0
LCORN	69069	56	42	63
	112063	**1**	14	67
	76715	37	50	14
MCORN	69069	27	40	39
	112063	80	62	66
	76715	86	62	9
ACORN	.10	24	72	**95**
	.81	63	48	11
	.12	6	47	27
ACORNI	69069	74	94	66
	112063	14	**2**	**97**
	76715	41	20	93
EXCEL	—	6	68	49
	—	41	84	14
	—	71	**95**	14
30000 Data				
Algorithm	Seed	0.0 - 1.0	0.0 - 0.1	0.9 - 1.0
LCORN	69069	**0**	**0**	**0**
	112063	**0**	**0**	**0**
	76715	**0**	**0**	**0**
MCORN	69069	38	31	40
	112063	60	73	16
	76715	24	40	25
ACORN	.10	9	66	86
	.81	92	80	7
	.12	16	76	24
ACORNI	69069	54	65	**96**
	112063	47	81	45
	76715	64	60	**98**
EXCEL	—	**98**	64	85
	—	72	**98**	88
	—	6	8	22
90000 Data				
Algorithm	Seed	0.0 - 1.0	0.0 - 0.1	0.9 - 1.0
LCORN	69069	**0**	**0**	**0**
	112063	**0**	**0**	**0**
	76715	**0**	**0**	**0**
MCORN	69069	31	12	**0**
	112063	**5**	48	87
	76715	44	40	76
ACORN	.10	23	**99**	60
	.81	96	35	8
	.12	44	79	11
ACORNI	69069	60	80	56
	112063	30	14	15
	76715	32	60	51
EXCEL	—	37	70	46
	—	33	11	94
	—	42	33	60

Table A.2: Results of uniformity test for 5 pseudo-random number generators and sequences of length 10000, 30000 and 90000.

10000 Data				
Algorithm	Seed	k = 3	k = 4	k = 5
LCORN	69069	**4**	**1**	30
	112063	**0**	**3**	11
	76715	**1**	**2**	**4**
MCORN	69069	47	70	28
	112063	51	**5**	6
	76715	34	82	60
ACORN	.10	55	83	35
	.81	6	**96**	20
	.12	41	18	90
ACORNI	69069	27	71	26
	112063	69	49	75
	76715	93	**99**	26
EXCEL	—	7	19	6
	—	10	33	78
	—	35	62	83
30000 Data				
Algorithm	Seed	k = 3	k = 4	k = 5
LCORN	69069	**0**	**0**	**0**
	112063	**0**	**0**	**0**
	76715	**0**	**0**	**0**
MCORN	69069	14	57	92
	112063	14	19	6
	76715	62	35	63
ACORN	.10	86	49	**98**
	.81	8	55	60
	.12	35	22	85
ACORNI	69069	12	82	41
	112063	54	55	89
	76715	**98**	20	56
EXCEL	—	41	82	61
	—	57	87	47
	—	87	41	55
90000 Data				
Algorithm	Seed	k = 3	k = 4	k = 5
LCORN	69069	**0**	**100**	**0**
	112063	**0**	**100**	**0**
	76715	**0**	**100**	**0**
MCORN	69069	47	63	59
	112063	11	53	7
	76715	65	41	83
ACORN	.10	13	28	57
	.81	69	59	76
	.12	78	**1**	21
ACORNI	69069	58	52	67
	112063	81	68	65
	76715	91	76	**99**
EXCEL	—	**97**	36	68
	—	66	71	62
	—	89	15	20

Table A.3: Results of k-dimensional uniformity test for 5 pseudo-random number generators and sequences of length 10000, 30000 and 90000.

10000 Data										
Algorithm	Seed	Runs	l=1	l=2	l=3	l=4	l=5	l=6	l=7	l=8
LCORN	69069	6638	4137	1809	549	121	19	3	0	0
	112063	**6568**	4053	1794	566	122	27	5	1	0
	76715	**6582**	4033	1861	539	122	23	4	0	0
MCORN	69069	6624	4115	1838	507	138	21	5	0	0
	112063	6642	4132	1827	543	119	18	3	0	0
	76715	6701	4219	1844	496	112	24	6	0	0
ACORN	.10	6702	4222	1829	508	121	21	1	0	0
	.81	6657	4166	1807	549	110	20	4	0	1
	.12	6691	4222	1798	528	120	22	0	1	0
ACORNI	69069	6630	4099	1879	502	119	27	3	1	0
	112063	6679	4176	1844	527	108	22	2	0	0
	76715	**6590**	4058	1847	530	123	28	3	1	0
EXCEL	—	6618	4103	1821	549	121	21	3	0	0
	—	6611	4087	1822	557	130	13	2	0	0
	—	6716	4235	1837	514	108	16	6	0	0
30000 Data										
Algorithm	Seed	Runs	l=1	l=2	l=3	l=4	l=5	l=6	l=7	l=8
LCORN	69069	**19817**	12305	5401	1656	371	67	15	1	1
	112063	**19863**	12307	5516	1592	371	65	10	1	1
	76715	**19834**	12307	5428	1660	355	69	14	1	0
MCORN	69069	19935	12399	5553	1534	372	62	11	4	0
	112063	19884	12319	5516	1634	341	63	10	1	0
	76715	19975	12443	5546	1569	341	63	13	0	0
ACORN	.10	**20157**	12707	5556	1467	362	60	4	1	0
	.81	20073	12612	5480	1578	335	58	8	1	1
	.12	20054	12616	5419	1615	327	71	5	1	0
ACORNI	69069	19905	12360	5548	1543	369	73	11	1	0
	112063	20047	12512	5606	1526	335	55	9	4	0
	76715	**19842**	12294	5482	1620	360	77	7	2	0
EXCEL	—	20036	12622	5408	1560	368	64	10	3	1
	—	20056	12533	5575	1545	342	54	6	1	0
	—	19946	12434	5500	1586	338	75	12	0	1
90000 Data										
Algorithm	Seed	Runs	l=1	l=2	l=3	l=4	l=5	l=6	l=7	l=8
LCORN	69069	**59580**	37010	16347	4892	1083	206	38	3	1
	112063	**59601**	37019	16399	4845	1092	204	37	3	2
	76715	**59512**	36887	16390	4908	1076	207	40	3	1
MCORN	69069	**59783**	37222	16498	4719	1137	172	29	6	0
	112063	59838	37274	16492	4799	1052	193	25	3	0
	76715	59989	37469	16543	4702	1070	173	31	1	0
ACORN	.10	**60387**	38167	16320	4652	1044	170	28	6	0
	.81	59954	37392	16620	4670	1040	201	26	4	1
	.12	59975	37504	16425	4814	990	215	21	6	0
ACORNI	69069	60045	37528	16565	4725	1006	187	31	3	0
	112063	59925	37383	16555	4696	1072	191	22	5	1
	76715	59816	37194	16548	4824	1037	193	16	4	0
EXCEL	—	59904	37277	16625	4770	1032	173	21	5	1
	—	60162	37850	16276	4820	995	176	38	7	0
	—	60060	37579	16559	4663	1027	192	35	5	0

Table A.4: Results of runs up and down test for 5 pseudo-random number generators and sequences of length 10000, 30000 and 90000.

10000 Data														
Algorithm	Seed	Runs	l=1	l=2	l=3	l=4	l=5	l=6	l=7	l=8	l=9	l=10	l=11	l=12
Theory		5000	2500	1250	625	313	156	78	39	20	10	5	2	1
LCORN	69069	4938	2450	1198	662	309	143	92	44	24	5	5	3	2
	112063	**4913**	2389	1249	644	292	179	81	39	21	10	4	1	3
	76715	**4904**	2408	1203	646	338	145	79	37	28	7	3	3	3
MCORN	69069	4970	2479	1266	576	310	169	93	40	17	8	4	6	2
	112063	4937	2413	1252	654	311	151	74	47	13	6	5	3	3
	76715	5061	2549	1309	591	293	167	72	46	20	5	6	2	1
ACORN	.10	5032	2543	1242	633	305	159	80	34	12	12	3	4	2
	.81	4981	2472	1243	638	335	135	77	34	31	6	4	2	3
	.12	5038	2548	1255	608	304	167	84	44	10	12	1	3	2
ACORNI	69069	5043	2513	1286	619	327	149	81	41	16	8	1	2	0
	112063	4937	2425	1251	615	337	142	84	50	8	10	9	4	0
	76715	5038	2558	1265	592	300	154	87	43	19	13	4	2	1
EXCEL	—	5038	2546	1265	582	334	163	87	28	14	8	3	3	4
	—	4980	2442	1320	585	318	165	71	36	21	11	7	1	0
	—	5067	2604	1229	623	309	144	81	32	21	12	8	2	1
30000 Data														
Algorithm	Seed	Runs	l=1	l=2	l=3	l=4	l=5	l=6	l=7	l=8	l=9	l=10	l=11	l=12
Theory		15000	7500	3750	1875	938	469	234	117	59	29	15	7	4
LCORN	69069	**14750**	7225	3688	1933	905	491	264	123	67	26	12	5	9
	112063	**14816**	7317	3706	1851	940	518	247	120	60	33	10	4	7
	76715	**14803**	7275	3697	1941	926	468	253	117	69	29	12	4	7
MCORN	69069	14859	7428	3646	1848	936	494	270	122	52	28	16	12	4
	112063	15049	7506	3828	1870	920	466	226	130	53	18	13	8	4
	76715	14936	7440	3813	1789	904	492	255	121	54	33	18	10	4
ACORN	.10	**15236**	7719	3799	1917	937	442	207	104	54	30	10	7	5
	.81	15046	7542	3789	1824	995	429	236	104	72	23	12	9	6
	.12	15077	7608	3724	1859	944	473	255	114	49	29	6	10	5
ACORNI	69069	15017	7419	3811	1941	943	455	246	104	46	29	8	8	3
	112063	15019	7540	3740	1867	924	469	242	134	40	28	23	4	4
	76715	14974	7481	3786	1829	938	424	261	125	68	37	15	3	5
EXCEL	—	15077	7610	3780	1805	942	482	217	119	62	26	16	8	8
	—	15017	7561	3743	1816	949	457	253	121	54	32	14	11	4
	—	14905	7404	3683	1914	961	474	223	134	57	29	13	7	5
90000 Data														
Algorithm	Seed	Runs	l=1	l=2	l=3	l=4	l=5	l=6	l=7	l=8	l=9	l=10	l=11	l=12
Theory		45000	22500	11250	5625	2813	1406	703	352	176	88	44	22	11
LCORN	69069	**44409**	21890	11067	5703	2813	1455	748	363	201	83	39	16	19
	112063	**44414**	21927	11073	5618	2829	1495	741	369	187	91	36	16	20
	76715	**44357**	21845	11052	5680	2853	1443	745	365	201	84	39	16	19
MCORN	69069	**44727**	22324	11087	5621	2783	1425	761	355	185	92	48	26	9
	112063	44901	22332	11351	5617	2772	1422	678	370	172	83	56	25	11
	76715	44942	22416	11326	5593	2772	1403	700	378	174	85	48	22	15
ACORN	.10	45147	22740	11163	5638	2829	1353	705	368	179	93	39	24	7
	.81	44779	22293	11173	5635	2824	1406	748	338	187	80	31	30	24
	.12	45138	22635	11360	5511	2774	1431	752	353	167	86	28	19	14
ACORNI	69069	45132	22577	11280	5688	2783	1442	701	340	160	98	23	20	10
	112063	45095	22675	11234	5595	2769	1377	730	364	162	87	62	15	14
	76715	44967	22365	11388	5627	2834	1308	716	371	175	103	39	17	15
EXCEL	—	44924	22321	11309	5729	2786	1391	691	345	171	87	40	21	19
	—	45227	22762	11211	5696	2780	1398	690	365	161	88	36	14	19
	—	**45256**	22760	11343	5607	2770	1375	704	348	177	93	40	22	7

Table A.5: Results of runs above and below the median test for 5 pseudo-random number generators and sequences of length 10000, 30000 and 90000.

10000 Data						
Algorithm	Seed	k = 1	k = 2	k = 3	k = 4	k = 5
LCORN	69069	56	**100**	**98**	**98**	**98**
	112063	**1**	**99**	**100**	**100**	**95**
	76715	37	**100**	85	**100**	**99**
MCORN	69069	27	54	73	37	**2**
	112063	80	29	78	85	8
	76715	86	78	70	85	66
ACORN	.10	24	18	39	44	83
	.81	63	91	94	**1**	10
	.12	6	19	64	93	71
ACORNI	69069	74	19	6	16	46
	112063	14	30	76	76	94
	76715	41	12	50	40	40
EXCEL	—	6	24	15	28	**5**
	—	41	48	14	70	68
	—	71	72	**96**	63	46
30000 Data						
Algorithm	Seed	k = 1	k = 2	k = 3	k = 4	k = 5
LCORN	69069	**0**	**100**	**100**	**100**	**100**
	112063	**0**	**100**	**100**	**100**	**100**
	76715	**0**	**100**	**100**	**100**	**100**
MCORN	69069	38	37	67	11	**0**
	112063	60	73	16	58	8
	76715	24	69	20	76	24
ACORN	.10	9	64	15	15	34
	.81	92	83	42	22	67
	.12	16	15	81	85	**96**
ACORNI	69069	54	**4**	73	24	60
	112063	47	81	91	**99**	**99**
	76715	64	26	41	9	57
EXCEL	—	**98**	67	55	90	27
	—	72	87	76	86	**99**
	—	6	**0**	22	43	18
90000 Data						
Algorithm	Seed	k = 1	k= 2	k = 3	k = 4	k = 5
LCORN	69069	**0**	**100**	**100**	**100**	**100**
	112063	**0**	**100**	**100**	**100**	**100**
	76715	**0**	**100**	**100**	**100**	**100**
MCORN	69069	31	18	28	7	24
	112063	5	38	**2**	22	28
	76715	44	93	20	57	11
ACORN	.10	23	72	11	**3**	**4**
	.81	**96**	66	25	19	62
	.12	44	79	88	55	29
ACORNI	69069	60	10	39	56	32
	112063	30	14	68	9	52
	76715	32	42	28	73	55
EXCEL	—	37	75	56	71	60
	—	33	86	86	79	82
	—	42	63	50	53	82

Table A.6: Results of maximum values test for 5 pseudo-random number generators and sequences of length 10000, 30000 and 90000.

A.4 Discussion

Pseudo-random generators are required for geostatistical simulation. Since truly random numbers cannot be generated by computer, we need to quantify the randomness of the pseudo-random sequences generated by different algorithms. Many different tests have been proposed to measure randomness, however, only a few of them are able to detect important departures from randomness.

Some powerful tests have been applied to five commonly used generators. They showed that the widely used `lcorn` generator does not give satisfactory results. Many applications that required random numbers a few years ago used this generator. We should test our pseudo-random number generators whenever a new powerful test is proposed. For the generators tested in this paper, `acorni` and `mcorn` performed the best, so they may be recommended as artifact-free pseudo-random number generators. `acorn` and the generator provided in `excel`, gave satisfactory results, but presented abnormal results more often than the previous two.

Test for randomness are of interest for geostatisticians, since they provide new ways to quantify correlation. The use of runs above thresholds is just one of them. Others could also be explored.

Appendix B

Exploratory Examples Using Runs

In this Appendix, the total number of runs above and below thresholds have been calculated for different correlated series. The results are compared with the theoretical distribution for uncorrelated sequences. Then, the frequencies of lengths of runs above thresholds are plotted in a map, along with the curve of average length for a given threshold. Again, different two-point variogram functions are used in order to see the differences with the random case. Thresholds have been chosen as regularly spaced quantiles.

Finally, maps of differences between the observed frequencies of lengths of runs in correlated sequences and the expected frequencies for the random case were plotted, showing again different responses given different two-point variogram functions.

B.1 Distribution of Total Number of Runs Above and Below Thresholds

Using `mcorn` (the generator with best results in tests documented in **Appendix 1**), 1,000 sequences of 10,000 pseudo-random numbers were generated. The number of runs above and below 4 thresholds were counted and compared with the theoretical limit distribution.

Histograms showing the distribution of total number of runs above and below the corresponding thresholds are shown in **Figure B.1**. The theoretical parameters of the distribution are summarized in **Table B.1** and compared with the observed ones. Both the mean and the standard deviation are close to their theoretical values.

1,000 Sequences of 10,000 Random Numbers - `mcorn`				
Threshold	Theoretical Mean	Theoretical Std. Dev.	Observed Mean	Observed Std. Dev.
0.2	3200	57.69	3199	59.20
0.4	4800	51.85	4800	51.65
0.6	4800	51.85	4799	51.97
0.8	3200	57.69	3200	58.32

Table B.1: Theoretical and observed results - `mcorn`.

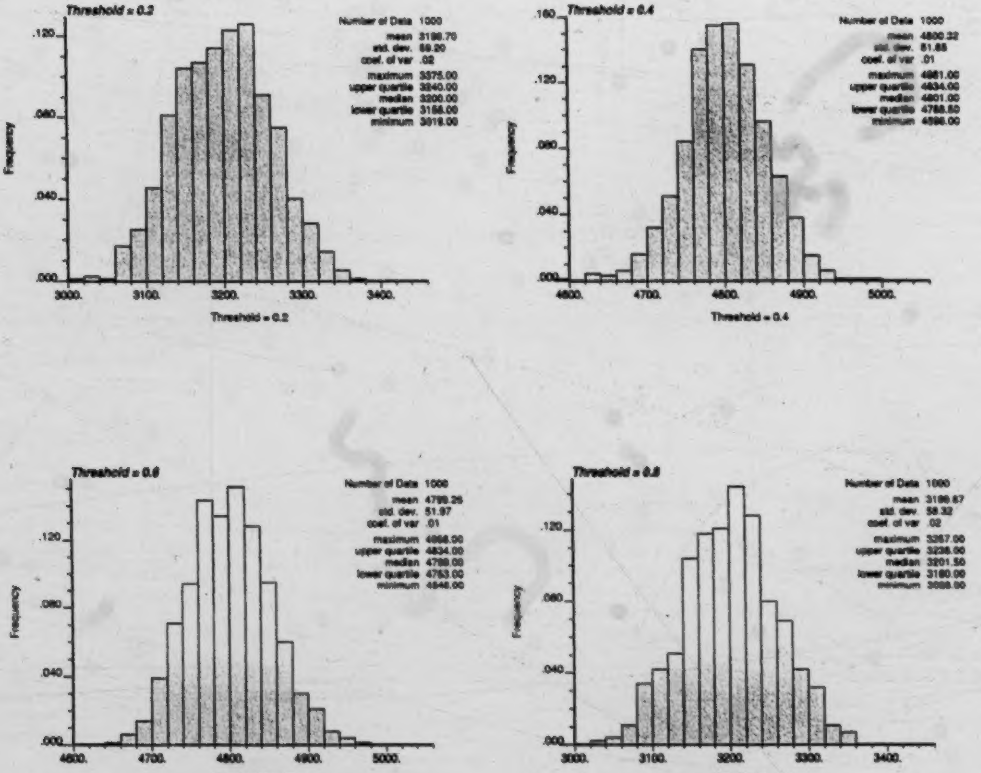


Figure B.1: Histograms of total number of runs for different thresholds - 1,000 sequences generated with `mcorn`.

B.2 Comparison of Different Variogram Functions

Recall that the distribution of runs of elements above and below a threshold (i.e. assumed independently drawn from a Bernoulli distribution with probabilities p and $q = 1 - p$, respectively) are asymptotically normally distributed with the following parameters:

$$\mu = 2 \cdot n \cdot p \cdot q \quad ,$$

$$\sigma^2 = 4 \cdot n \cdot p \cdot q \cdot (1 - 3 \cdot p \cdot q)$$

Two of the random number generators were compared with the expected number of runs (for uncorrelated values). `mcorn` and `acorni` showed a very good reproduction of the theoretical mean, as presented in **Figure B.2**. The standard deviation is not as smooth as the mean, but notice the good reproduction at extremes; this is common to all cases presented. The mean and standard deviation of the total number of runs above and below each threshold was calculated as an average over 100 sequences of 1000 values each.

Series of correlated data were generated using moving average simulation and simulated annealing. The first example considers a triangular variogram function (this variogram model is valid in one dimension only):

$$\gamma(\mathbf{h}) = \begin{cases} h, & \text{if } h \leq a \\ a, & \text{if } h > a \end{cases}$$

A wide variety of ranges were evaluated using sequences generated by moving average (**Figure B.3**). The curves of mean and standard deviation of the total number of runs depart predictably from the uncorrelated case. When correlation increases, runs tend to be longer, so there are less than in the random case. For some ranges (5, 10, 15, 20, and 25 units) simulated annealing was used to generate correlated series. **Figure B.4** gives the result for a triangular variogram function. Some different variogram models were explored with similar results: in all the cases, the mean number of runs decreases when the sequence has a greater correlation range.

Three different seed numbers were used with a fixed range (equals to 5 units). The results showed that there is no significant differences between the sequences generated with different seed numbers. Notice that for every sequence in those examples a different seed number was used.

In general, the curve of mean total number of runs is quite smooth and well behaved, however, the standard deviation does not seem to be as stable as the mean. Differences between moving average and simulated annealing might be due to the random function implicit in each method. In the first case, Gaussianity is derived from the averages and the Central Limit Theorem. In the case of simulated annealing, the random function is unknown. In order to visualize the differences between different variogram models and between the methods used to generate the sequences, **Figure B.5** is presented comparing the result for a correlation range of 5 units. The theoretical result for uncorrelated sequences is plotted as a reference. The same comparison was done for other ranges. An interesting and consistent difference between the results given by moving average and simulated annealing (using

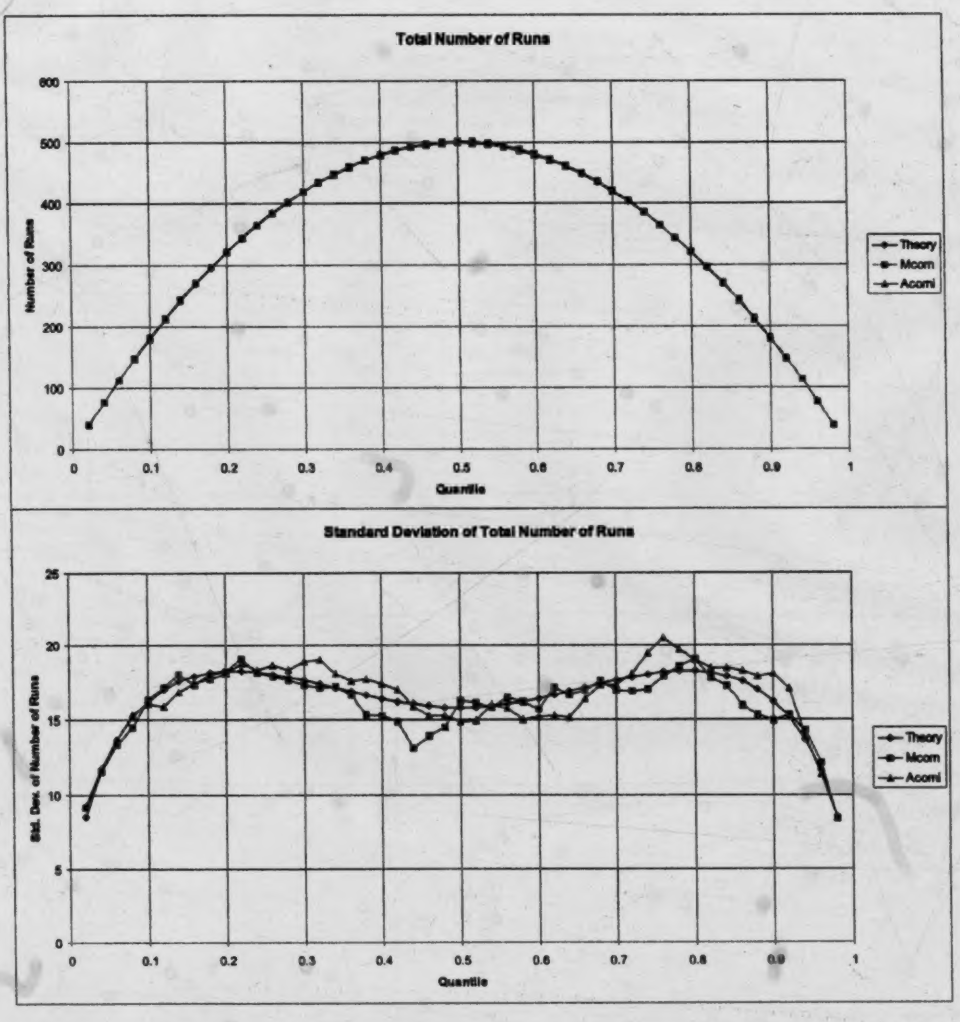


Figure B.2: Mean and standard deviation of total number of runs above and below thresholds (quantiles) for `mcorn` and `acorni`, compared with the theoretical expected values.

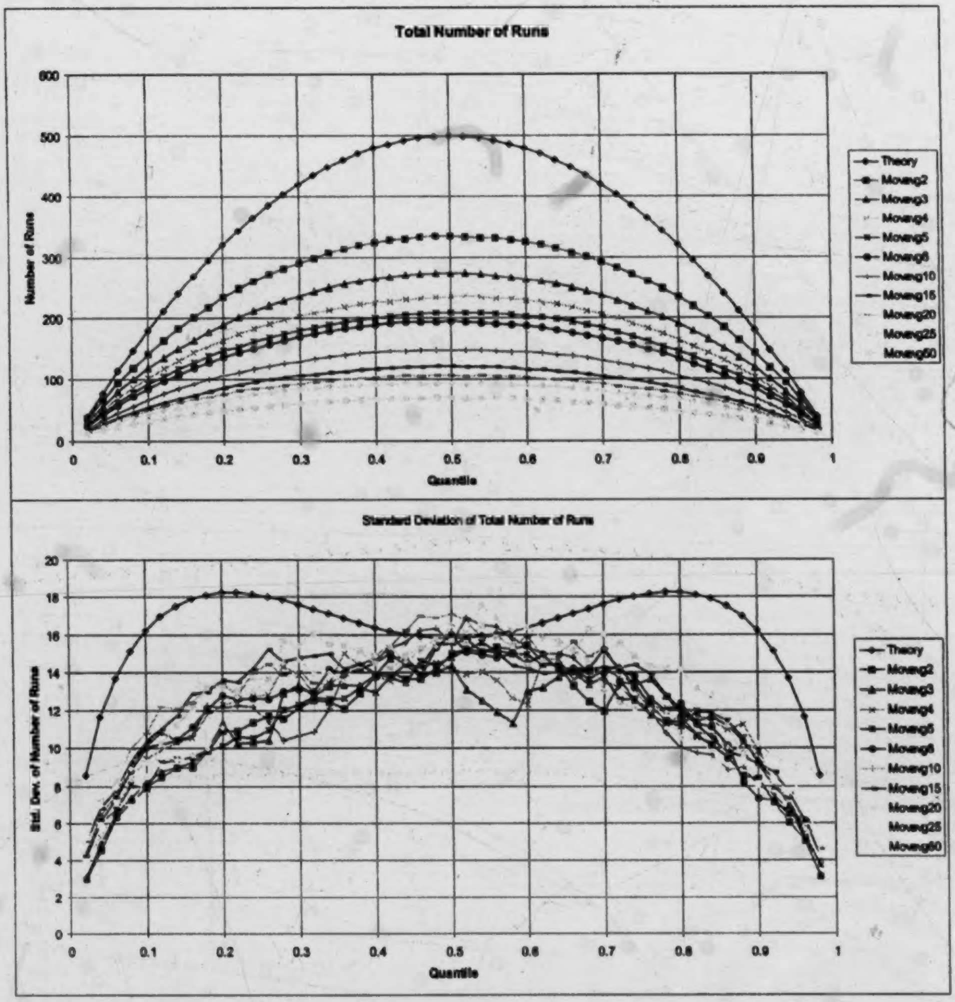


Figure B.3: Mean and standard deviation of total number of runs above and below thresholds (quantiles) for sequences with a triangular variogram function generated using moving average, compared with the theoretical expected values for uncorrelated series.

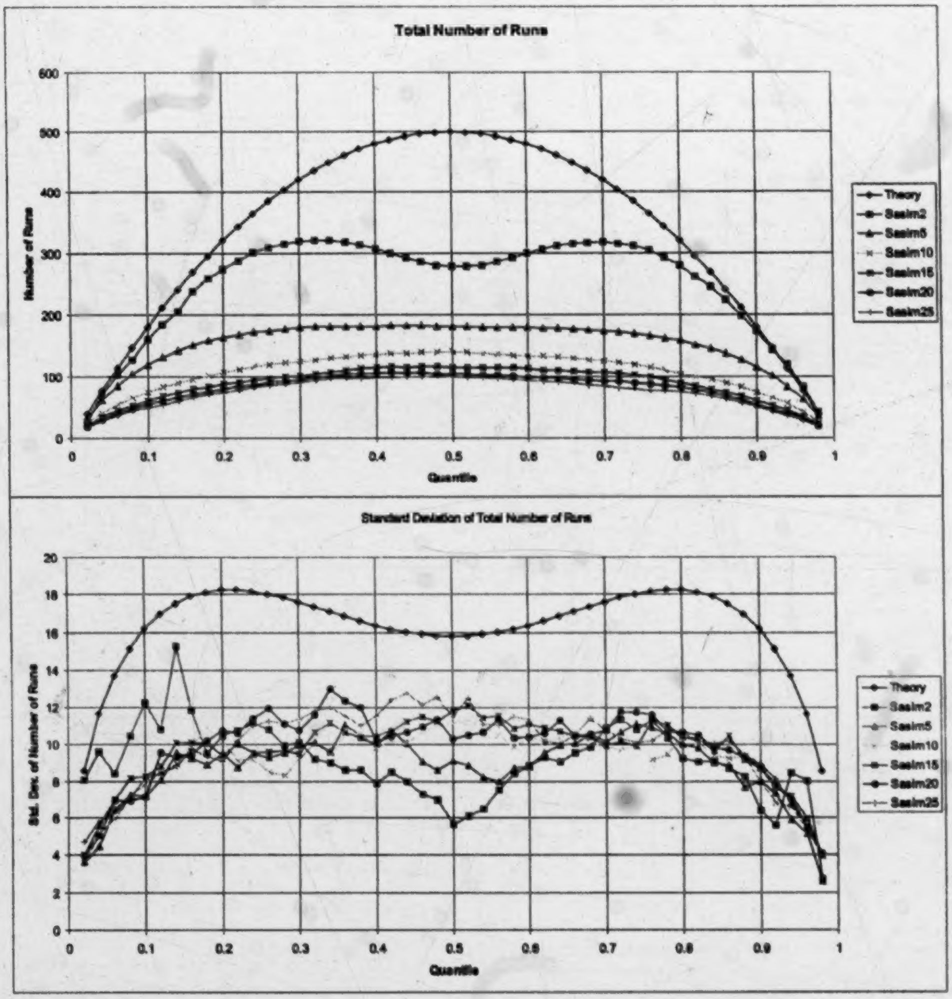


Figure B.4: Mean and standard deviation of total number of runs above and below thresholds (quantiles) for sequences with a triangular variogram function generated using simulated annealing, compared with the theoretical expected values for uncorrelated series.

the triangular variogram function) is demonstrated here. In all the cases the moving average method (Gaussian) generates standard deviations closer to the random case than the simulated annealing technique. This situation can be explained by the maximum entropy property of the Gaussian model. Differences between the triangular, spherical and exponential variogram are due to the different correlation for a given distance, as presented in **Figure B.6**.

B.3 Maps of Frequencies of Length of Runs Above Each Threshold

In order to obtain a plot easily understandable and that reflects clearly any change in the high order behavior of the variable, a map of lengths of frequencies of lengths of runs above each threshold along with a curve showing the average length of runs above each threshold has been implemented.

Using sequences with ranges of 2, 5, 10, 15, 20, and 25 units, the number of runs above each threshold were calculated. The decision of using only the runs above (instead of runs above and below) was taken because using both might hide differences in the continuity of high and low values, by averaging the number of runs.

Figure B.7 shows the maps for random sequences generated with `acorni` and `mcorn`. **Figure B.8** shows the maps for sequences generated using moving average with a triangular variogram model. In **Figure B.9** sequences with the same variogram model were generated using simulated annealing. Some other examples with different correlation functions are not presented here.

In all the cases, when the range increases, the cloud of non zero frequencies grows to the right and up, because when the range of correlation is greater, long runs are more likely to be found.

When different models of correlation are used, slight differences in the cloud of frequencies can be seen. The curve of average lengths also changes when different variogram models are used.

The next section presents another way to look at high order correlation. Subtracting the expected frequencies of lengths of runs for the random case to the observed frequencies, maps of differences were generated.

B.4 Maps of Differences in Frequencies of Lengths of Runs

The expected number of runs above a threshold of a given length i for a random sequence, r_{1i}, can be expressed as [123]:

$$E(r_{1i}) = \frac{(n_2+1)^{(2)} \cdot n_1^{(i)}}{n^{(i+1)}}$$

where $n_1 = n \cdot p_1$, $n_2 = n \cdot p_2$, and $x^{(a)} = x \cdot (x-1) \cdot \ldots \cdot (x-a+1)$

The difference between frequencies observed from correlated sequences and the expected for the random case were calculated. **Figure B.10** shows the maps of differences using the pseudo-random sequences generated using `acorni` and `mcorn`.

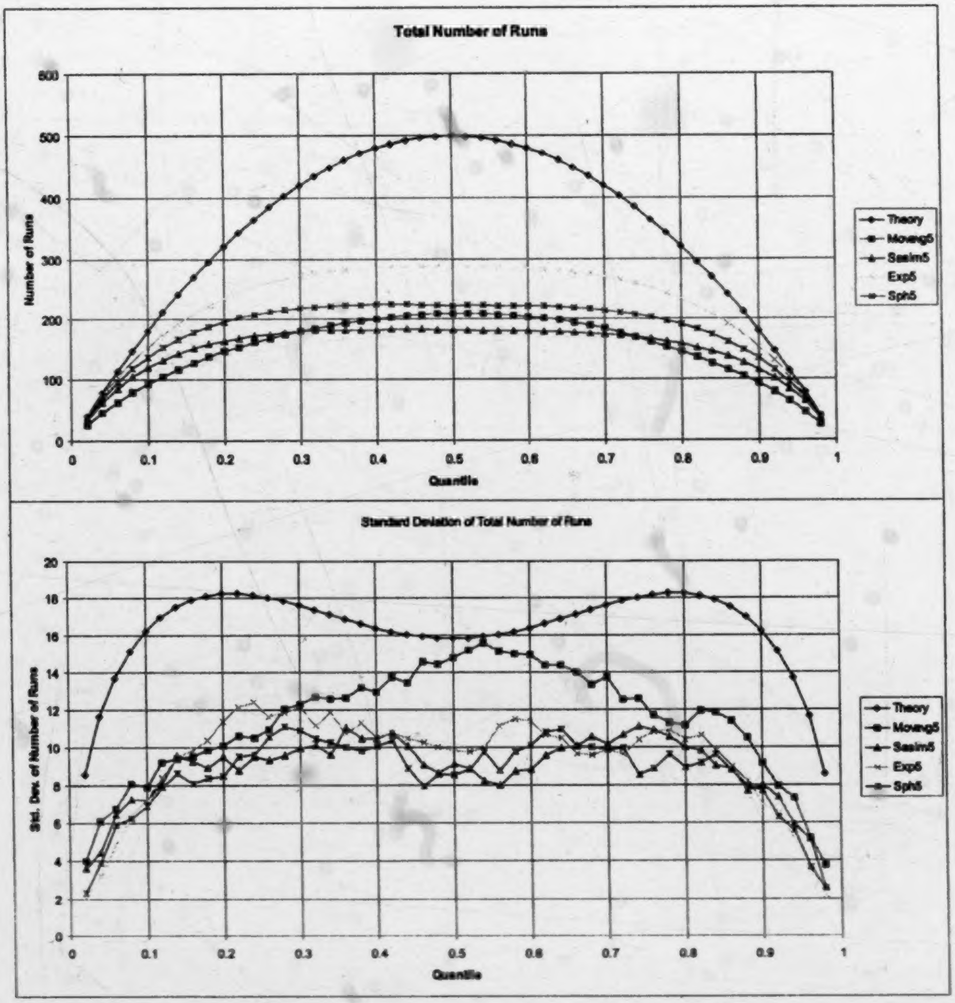


Figure B.5: Mean and standard deviation of total number of runs above and below thresholds (quantiles) for sequences with a range of 5 and different variogram functions (`sasim` and `movavg` have a triangular variogram).

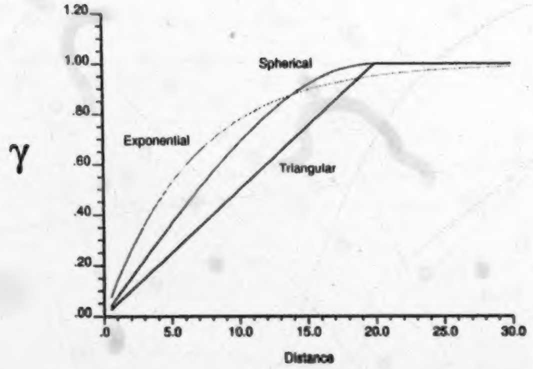


Figure B.6: Variogram models used in the examples (relative shape for effective range of 20).

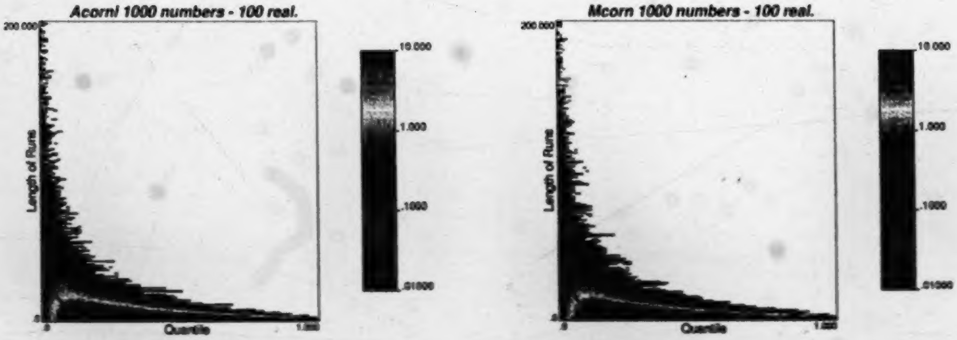


Figure B.7: Map of frequency of lengths of runs above quantiles for random sequences generated with `acorni` and `mcorn`. The solid line shows the average length as a function of the quantile.

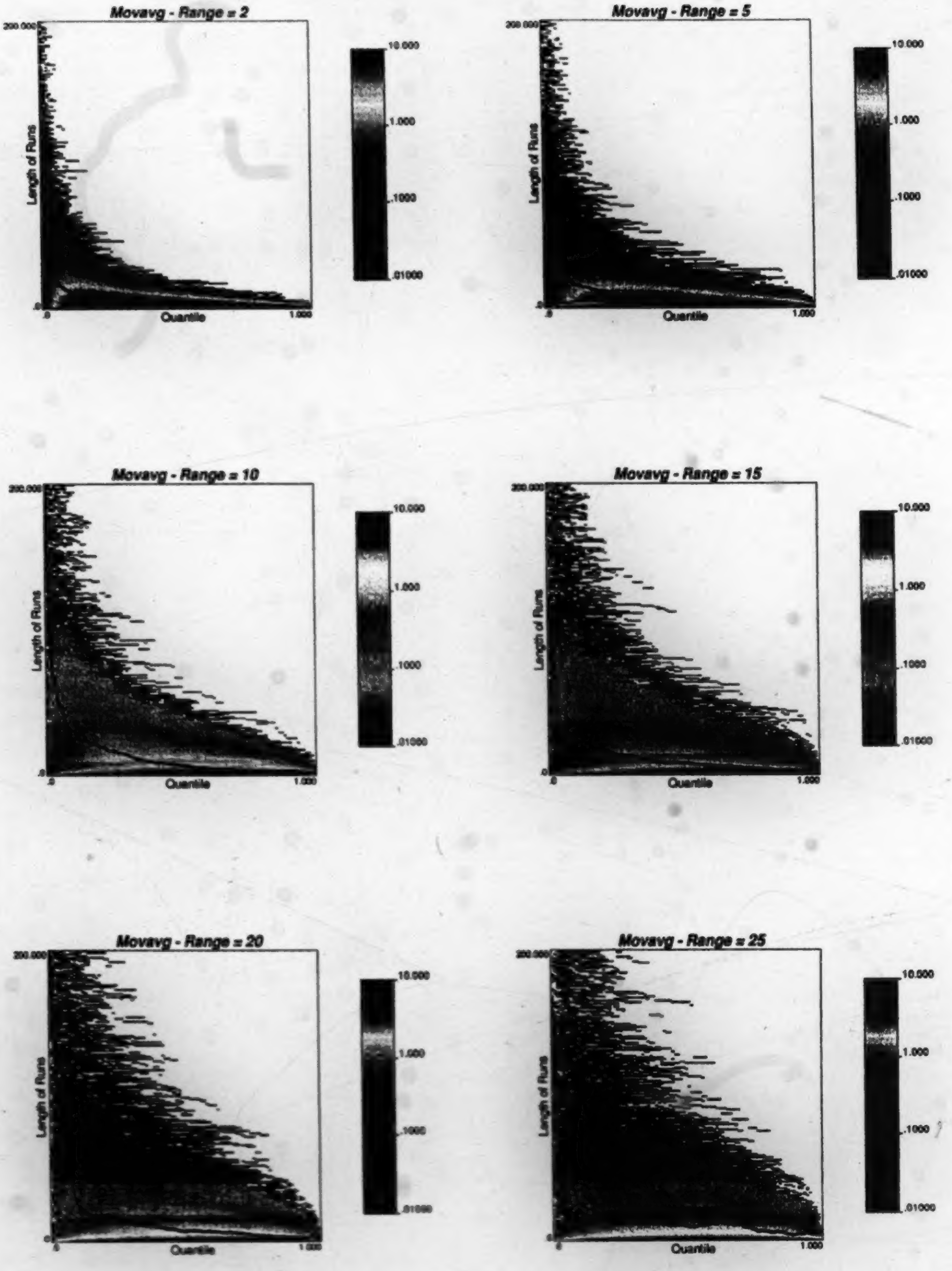
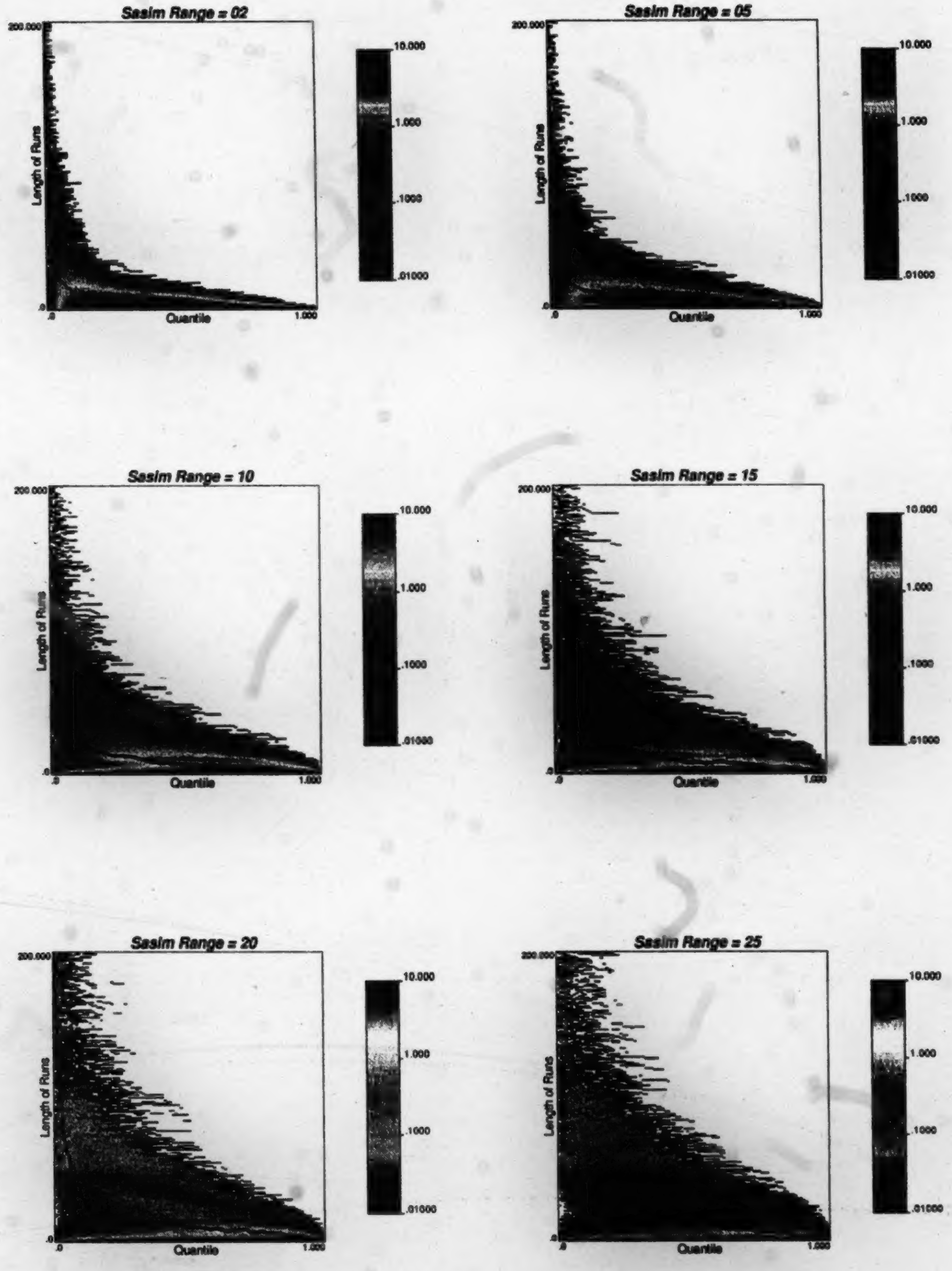


Figure B.8: Map of frequency of lengths of runs above quantiles for sequences generated by moving average (triangular variogram model). The solid line shows the average length as a function of the quantile.


Sasim Range = 02

Sasim Range = 05

Sasim Range = 10

Sasim Range = 15

Sasim Range = 20

Sasim Range = 25

Figure B.9: Map of frequency of lengths of runs above quantiles for sequences generated by simulated annealing (triangular variogram model). The solid line shows the average length as a function of the quantile.

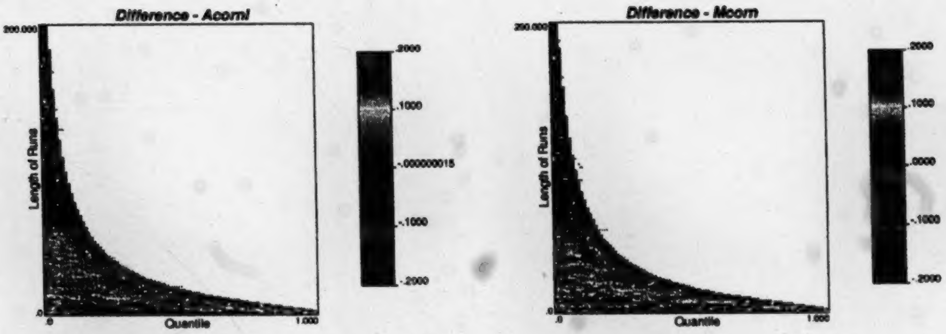


Figure B.10: Map of differences of frequencies of lengths of runs above quantiles for sequences generated with `acorni` and `mcorn`.

This just shows that the pseudo-random number generators do not depart significantly from the theoretical values. **Figure B.11** shows the maps of differences for sequences generated using moving average with a triangular variogram model. The differences for the same variogram model, but for sequences generated with simulated annealing are presented in **Figure B.12**. Again, different correlation functions were used with similar results and are not shown in this Appendix. A characteristic zone where the observed frequencies are lower than the expected ones is repeatedly seen for all ranges and variogram models; there are fewer short runs. Then, there is a zone where the observed frequencies are higher than the expected for the random case: there are more longer runs. In other words, when correlated sequences are used, there are less short runs and more long ones than in the random case.

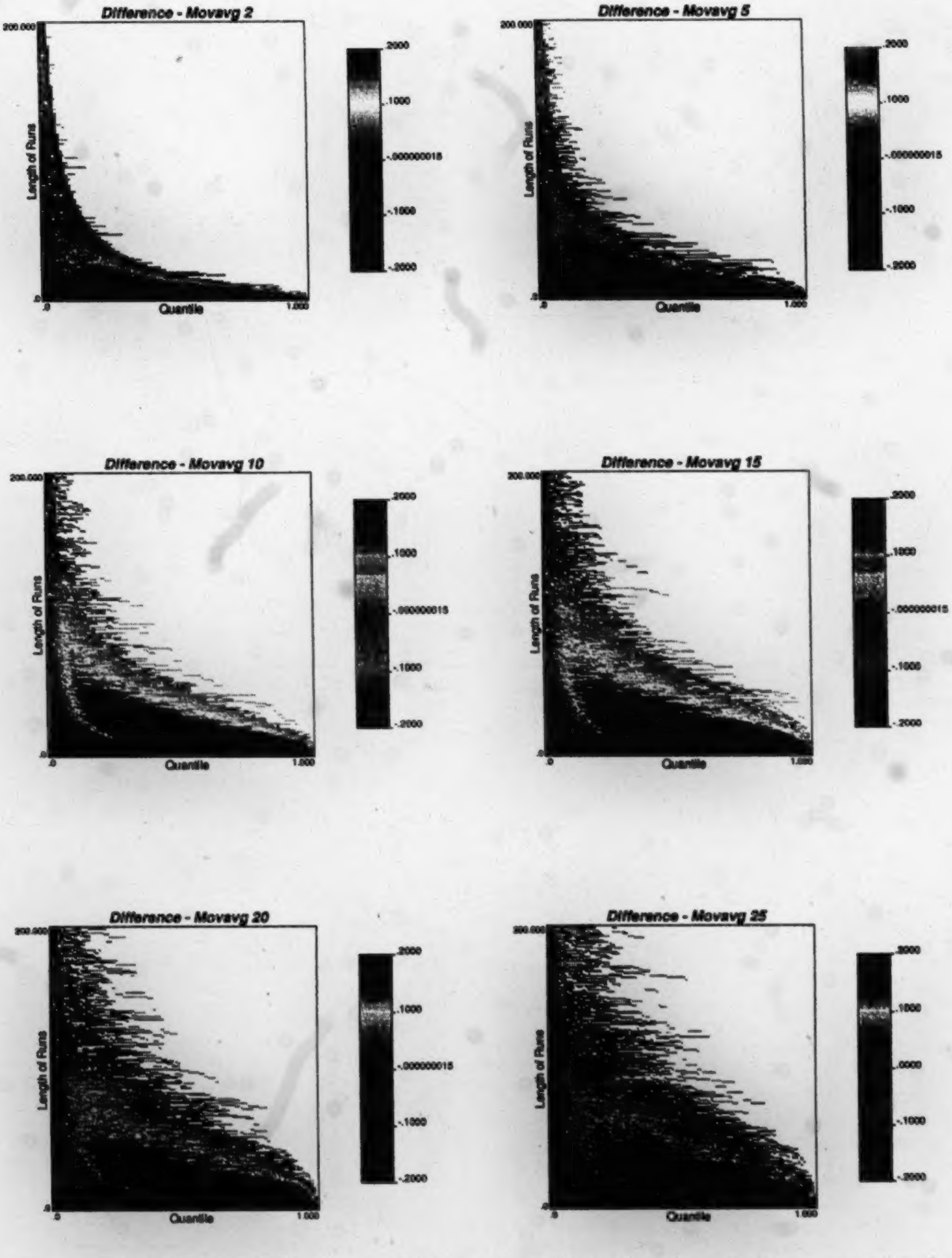


Figure B.11: Map of differences of frequencies of lengths of runs above quantiles for sequences generated by moving average (triangular variogram model).

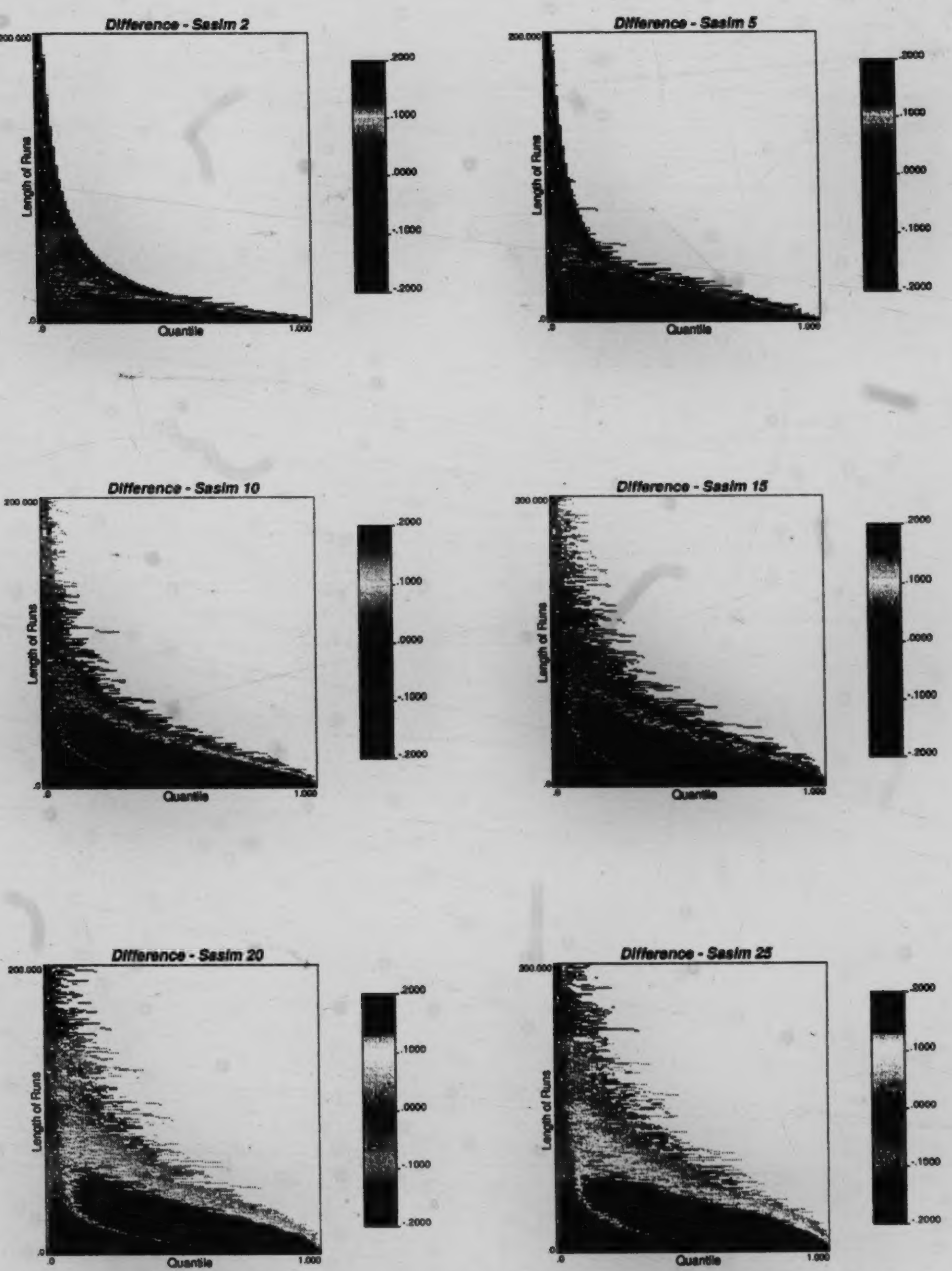


Figure B.12: Map of differences of frequencies of lengths of runs above quantiles for sequences generated by simulated annealing (triangular variogram model).

Appendix C

Calculation of Uncertainty in the Variogram

There are often limited data available in early stages of geostatistical modeling. This leads to considerable uncertainty in statistical parameters including the variogram. This Appendix presents an approach to calculate the uncertainty in the variogram. A methodology to transfer this uncertainty through geostatistical simulation and decision making is also presented.

The experimental variogram value $2\hat{\gamma}(\mathbf{h})$ for a separation lag vector $\mathbf{h}$ is a mean of squared differences. The variance of a mean can be calculated with a model of the correlation between the pairs of data used in the calculation. The "data" here are squared differences; therefore, we need a measure of four-point correlation. A theoretical multivariate Gaussian approach is presented for this uncertainty assessment together with a number of examples. The theoretical results are validated by numerical simulation. The simulation approach permits generalization to non-Gaussian situations.

Multiple plausible variograms may be fit knowing the uncertainty at each variogram point, $2\gamma(\mathbf{h})$. Multiple geostatistical realizations may then be constructed and subjected to process assessment to measure the impact of this uncertainty.

C.1 Introduction

Variogram modelling is a critical step in any geostatistical study; however, a reliable variogram is difficult to infer in presence of sparse data. This is particularly true in the early exploration stages of an ore deposit or petroleum reservoir. A quantitative model of the uncertainty in the variogram would allow an assessment of uncertainty from geostatistical simulation.

Notwithstanding robust procedures to calculate variograms and other measures of spatial correlation [25, 24, 61] there is unavoidable uncertainty in the variogram. There are many references on the calculation and use of the variogram (including [70, 129, 130]); however, there is little on the calculation of the unavoidable uncertainty in the variogram.

We show how to calculate the pointwise uncertainty in the variogram. This pointwise uncertainty must be translated to joint uncertainty, that is, into uncertainty in the variogram *model*. Within the bounds of pointwise uncertainty, we propose to establish different scenarios, ranging from small continuity to great continuity. These "scenarios" can be used to evaluate the consequences of the choice of the variogram model after simulating a number of realizations of the transfer function. These realizations can be used to determine the sensitivity of the results to variogram uncertainty.

C.2 Pointwise Variogram Uncertainty

The variogram is defined as:

$$2 \cdot \gamma(\mathbf{h}) = Var\{Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})\} \tag{C.1}$$

where $Z(\cdot)$ is an element of a random field $\{Z(\mathbf{u}) : \mathbf{u} \in D\}$.

A method of moments estimator of the variogram $2\gamma(\mathbf{h})$ is the average of squared differences between data separated exactly by that distance vector $\mathbf{h}$ (in practice, we define angle and lag tolerances, so that $n(\mathbf{h})$ is the number of pairs approximately $\mathbf{h}$ appart):

$$2 \cdot \hat{\gamma}(\mathbf{h}) = \frac{1}{n(\mathbf{h})} \cdot \sum_{i=1}^{n(\mathbf{h})} [Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2 \tag{C.2}$$

where in practice $n(\mathbf{h})$ is the number of data pairs approximately $\mathbf{h}$ apart.

Consider $X_i = [Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2$, the squared difference between the values at locations $\mathbf{u}_i$ and $\mathbf{u}_i + \mathbf{h}$. The variogram is the mean of the X_i's:

$$\bar{X} = 2 \cdot \hat{\gamma}(\mathbf{h}) = \frac{1}{n(\mathbf{h})} \cdot \sum_{i=1}^{n(\mathbf{h})} X_i \tag{C.3}$$

From classical statistics, we know that the uncertainty in the mean $\bar{X}$ is defined as:

$$Var\{\bar{X}\} = E\{(\bar{X} - E\{\bar{X}\})^2\} = E\{\bar{X}^2\} - (E\{\bar{X}\})^2 \tag{C.4}$$

Now, using **Equation C.4** we can calculate the uncertainty in the variogram assuming that we have a "reference" variogram model fitted to the experimental points. $\bar{X}$ is replaced by $2 \cdot \hat{\gamma}(\mathbf{h})$ and the variance of squared differences around the model is calculated as follows:

$$\begin{aligned} \sigma^2_{2\cdot\hat{\gamma}(\mathbf{h})} &= E\{(2 \cdot \hat{\gamma}(\mathbf{h}))^2\} - (E\{2 \cdot \hat{\gamma}(\mathbf{h})\})^2 \\ &= E\{(\tfrac{1}{n(\mathbf{h})} \cdot \textstyle\sum_{i=1}^{n(\mathbf{h})} [Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2)^2\} - (E\{2 \cdot \hat{\gamma}(\mathbf{h})\})^2 \\ &= E\{(\tfrac{1}{n(\mathbf{h})^2} \cdot \textstyle\sum_{i=1}^{n(\mathbf{h})} \sum_{j=1}^{n(\mathbf{h})} [Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2 \cdot [Z(\mathbf{u}_j) - Z(\mathbf{u}_j + \mathbf{h})]^2\} \\ &\quad -(2 \cdot \hat{\gamma}(\mathbf{h}))^2 \end{aligned} \tag{C.5}$$

$$\sigma^2_{2\cdot\hat{\gamma}(\mathbf{h})} = \frac{1}{n(\mathbf{h})^2} \cdot \sum_{i=1}^{n(\mathbf{h})} \sum_{j=1}^{n(\mathbf{h})} E\{[Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2 \cdot [Z(\mathbf{u}_j) - Z(\mathbf{u}_j + \mathbf{h})]^2\} - (2 \cdot \hat{\gamma}(\mathbf{h}))^2 \tag{C.6}$$

This can be simplified by using the definition of the covariance:

$$\begin{aligned} C_{ij}(\mathbf{h}) &= Cov\{X_i, X_j\} = E\{(X_i - E\{X_i\}) \cdot (X_j - E\{X_j\})\} \\ &= E\{X_i \cdot X_j\} - E\{X_i\} \cdot E\{X_j\} \\ &= E\{X_i \cdot X_j\} - \bar{X}^2 \end{aligned} \tag{C.7}$$

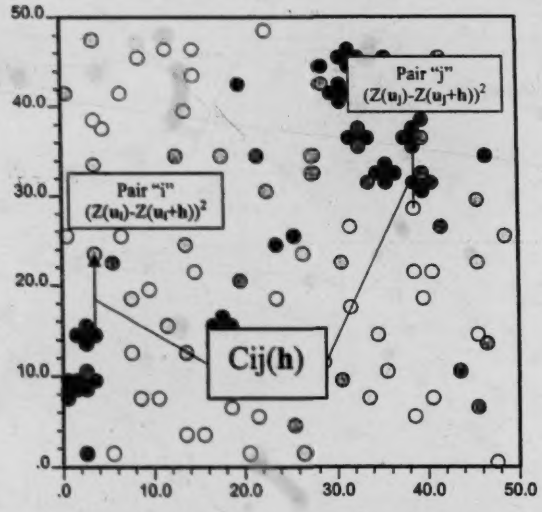


Figure C.1: Calculation of fourth order covariances $C_{ij}(\mathbf{h})$. For a given lag vector $\mathbf{h}$, the fourth order covariance corresponds to the covariance between the squared differences of pairs i and j.

Now, replacing X_i and X_j by the squared differences $[z(\mathbf{u}_i) - z(\mathbf{u}_i + \mathbf{h})]^2$ and $[z(\mathbf{u}_j) - z(\mathbf{u}_j + \mathbf{h})]^2$ respectively, and $\bar{X}$ by the variogram $2 \cdot \gamma(\mathbf{h})$:

$$C_{ij}(\mathbf{h}) = E\{[Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2 \cdot [Z(\mathbf{u}_j) - Z(\mathbf{u}_j + \mathbf{h})]^2\} - (2 \cdot \hat{\gamma}(\mathbf{h}))^2 \qquad \text{(C.8)}$$

A simple formula for the variance of a particular variogram value is obtained replacing the covariance (**Equation C.8**) in **Equation C.6**:

$$\sigma^2_{2\cdot\hat{\gamma}(\mathbf{h})} = \frac{1}{n(\mathbf{h})^2} \cdot \sum_{i=1}^{n(\mathbf{h})} \sum_{j=1}^{n(\mathbf{h})} C_{ij}(\mathbf{h}) \qquad \text{(C.9)}$$

where $C_{ij}(\mathbf{h})$ is calculated as in **Equation C.8**. To avoid confusion, note that $C_{ij}(\mathbf{h})$ is covariance between pair i $[Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2$ and j $[Z(\mathbf{u}_j) - Z(\mathbf{u}_j + \mathbf{h})]^2$ (**Figure C.1**).

Equation C.9 tells us that the uncertainty in the variogram at a distance $\mathbf{h}$ is the average covariance between "pairs of pairs" used to calculate the variogram for that particular lag.

The covariance between "pairs of pairs" can be calculated theoretically under a multiGaussian assumption. The following section presents this approach. The next sections present the Local and Global Simulation Methods to check the results given by the Theoretical Approach. The Global Simulation Method is more general in the sense that it gives the whole distribution of uncertainty in the variogram values for each lag. Although the shape of the pointwise uncertainty distribution is unknown and we know that the variogram values must be non-negative, a Gaussian shape was assumed to present the confidence intervals calculated using the variance in the theoretical approach and the local simulation method. Theory says that if all the squared random variables are independent (which is clearly not the case) the distribution of uncertainty in a variogram point should be χ^2 (chi square). The Global Simulation Method shows in few cases asymmetric distributions; however, a Gaussian distribution is a good approximation in most of the cases.

The following steps are required for all three methodologies,

1. Transform data to normal space: Any data distribution can be easily transformed to a Gaussian univariate distribution. In the following examples the program `nscore` in GSLIB [39] was used to perform the transformation. This transformation is commonly done to allow Gaussian simulation.

2. Check multivariate Gaussianity: To fulfil the multivariate Gaussian condition, one should assure that not only the univariate distribution is Gaussian, but also the bivariate and all multivariate distributions. In practice, some tests can be done to the transformed distribution in order to accept bi-Gaussianity; however, they are not often applied, especially in presence of sparse data.

3. Calculate the experimental variogram: The location of the sampled points and the values of the variable under study at these locations are used to calculate the experimental variogram, $2 \cdot \hat{\gamma}(\mathbf{h})$.

4. Fit a variogram model: The fitted variogram model is critical for subsequent stages of uncertainty evaluation. The requirement for a variogram model to assess uncertainty in the variogram is of some concern. Nevertheless, a model assumption is required to proceed.

The difference between the Theoretical Approach and the Numerical methods lies in how the variance for each lag is calculated.

C.3 Theoretical Approach

Assuming that the regionalized variable is multivariate Gaussian the variogram uncertainty can be calculated from theory. Expanding **Equation C.8**, the covariance can be written as a sum of fourth order moments:

$$\begin{aligned}
C_{ij}(\mathbf{h}) &= E\{[Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2 \cdot [Z(\mathbf{u}_j) - Z(\mathbf{u}_j + \mathbf{h})]^2\} - (2 \cdot \hat{\gamma}(\mathbf{h}))^2 \\
&= E\{Z(\mathbf{u}_i)^2 \cdot Z(\mathbf{u}_j)^2 \\
&\quad -2 \cdot Z(\mathbf{u}_i)^2 \cdot Z(\mathbf{u}_j) \cdot Z(\mathbf{u}_j + \mathbf{h}) + Z(\mathbf{u}_i)^2 \cdot Z(\mathbf{u}_j + \mathbf{h})^2 \\
&\quad -2 \cdot Z(\mathbf{u}_i) \cdot Z(\mathbf{u}_i + \mathbf{h}) \cdot Z(\mathbf{u}_j)^2 \\
&\quad +4 \cdot Z(\mathbf{u}_i) \cdot Z(\mathbf{u}_i + \mathbf{h}) \cdot Z(\mathbf{u}_j) \cdot Z(\mathbf{u}_j + \mathbf{h}) \\
&\quad -2 \cdot Z(\mathbf{u}_i) \cdot Z(\mathbf{u}_i + \mathbf{h}) \cdot Z(\mathbf{u}_j + \mathbf{h})^2 + Z(\mathbf{u}_i + \mathbf{h})^2 \cdot Z(\mathbf{u}_j)^2 \\
&\quad -2 \cdot Z(\mathbf{u}_i + \mathbf{h})^2 \cdot Z(\mathbf{u}_j) \cdot Z(\mathbf{u}_j + \mathbf{h}) + Z(\mathbf{u}_i + \mathbf{h})^2 \cdot Z(\mathbf{u}_j + \mathbf{h})^2\} \\
&\quad -(2 \cdot \hat{\gamma}(\mathbf{h}))^2
\end{aligned} \tag{C.10}$$

This covariance is called a *quadratic covariance* [117] and it can be calculated if $Z(\mathbf{u}_i)$, $Z(\mathbf{u}_i+\mathbf{h})$, $Z(\mathbf{u}_j)$, and $Z(\mathbf{u}_j+\mathbf{h})$ have a multivariate Gaussian distribution. In such case, any fourth order moment can be calculated using the pairwise covariance values as follows:

$$E\{Z_1 \cdot Z_2 \cdot Z_3 \cdot Z_4\} = C_{12} \cdot C_{34} + C_{13} \cdot C_{24} + C_{14} \cdot C_{23} \tag{C.11}$$

Notice that those pairwise covariances are different than the $C_{ij}(\mathbf{u})$ presented earlier, which are fourth order statistics, since they correspond to the covariance between pairs of squared differences (i.e. "pairs of pairs").

Then, the variogram variance is calculated as a sum of fourth order moments minus two times the variogram squared.

A simple program can perform these calculations. For each lag, the location of pairs considered in the experimental variogram calculation is used to determine the fourth order moment as follows:

$$\begin{aligned} E\{z(\mathbf{u}_i)\cdot z(\mathbf{u}_i+\mathbf{h})\cdot z(\mathbf{u}_j)\cdot z(\mathbf{u}_j+\mathbf{h})\} &= C(z(\mathbf{u}_i), z(\mathbf{u}_i+\mathbf{h}))\cdot C(z(\mathbf{u}_j), z(\mathbf{u}_j+\mathbf{h})) \\ &+C(z(\mathbf{u}_i), z(\mathbf{u}_j))\cdot C(z(\mathbf{u}_i+\mathbf{h}), z(\mathbf{u}_j+\mathbf{h})) \\ &+C(z(\mathbf{u}_i), z(\mathbf{u}_j+\mathbf{h}))\cdot C(z(\mathbf{u}_i+\mathbf{h}), z(\mathbf{u}_j)) \end{aligned}$$

A valid, positive definite, covariance model is required to perform the calculation presented above. That is the reason to require a first guess of the variogram model.

C.4 Simulation Alternative

C.4.1 Local Simulation Method

The idea is to simulate each set of four-point locations in turn and evaluate the fourth order moments in **Equation C.10** by simple averages. Again, the assumption of multivariate Gaussianity simplifies the simulation. A matrix or LU simulation approach is very fast and efficient since only four points are considered at a time and there are no conditioning data. All fourth order moments in **Equation C.10** are estimated as averages of products using the simulated values, and the variogram variance is calculated with **Equation C.9**.

C.4.2 Global Simulation Method

The basic idea is to generate non-conditional realizations of the domain using the variogram model, and then calculate the variogram using only the values at the sampled locations. The variance between the variogram values at each lag calculated using these realizations should converge to the same value obtained through any of the other approaches; however, the advantage of this approach is that we can estimate the entire uncertainty distribution of all variogram lags simultaneously, without assuming its shape.

This approach was implemented using the GSLIB program `sgsim`, that is, unconditional realizations are generated. The sequential path in the program could be modified to only simulate the locations of the original data. Uncertainty in the variogram is directly evaluated by the variability between multiple realizations.

This global simulation method can be viewed as a "spatial bootstrap" or resampling from geostatistical realizations [89].

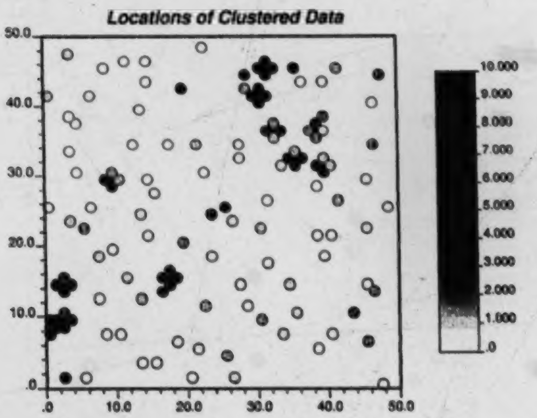


Figure C.2: Location map of samples taken from Cluster database.

C.5 Validation of Theoretical Approach by Simulation

The Theoretical Approach has the following advantages over the two simulation-based methods (1) implementation is easier since the fourth order moments are calculated analytically and directly, (2) computer speed is much improved since there is no need for random number generation or multiple realizations, (3) the simulation methods are approximate, although they converge to the correct result.

The Global Simulation Method has the advantage that the entire distribution of uncertainty is simulated.

C.6 Example 1: Cluster.dat

Consider the database `cluster.dat` available in GSLIB [39]. The sample locations are in a pseudo-regular grid, with clusters in the high value zones (**Figure C.2**). After normal score transformation, the north-south variogram is calculated for five lags, using a lag separation distance of 4.0 and a lag tolerance of 2.0.

An isotropic spherical variogram model with range 15 m and 90 % of variance contribution is fitted to the experimental variogram. The nugget effect is 0.1 (10 % of variance contribution):

$$\gamma(\mathbf{h}) = 0.1 + 0.9 \cdot Sph(\frac{h}{15}) \tag{C.12}$$

The variogram uncertainty is assessed theoretically, using local simulation, and through the global simulation method. The variance has been calculated for each lag using the three methodologies presented above. In the local simulation approach (using LU simulation), 100 realizations were performed. The results are presented in **Table C.1**.

Results show that with a reasonable number of LU simulations, the Local Simulation Method gives a variance very close to the theoretical result. Assuming normality in the uncertainty distribution, the confidence intervals can be calculated.

Lag	Lag Distance	Experimental Variogram	Fitted Variogram	Variance Theo. App.	Var. Local Sim. Meth.	Var. Global Sim. Meth.
2	1.395	0.262	0.225	0.004	0.004	0.004
3	4.361	0.431	0.481	0.021	0.019	0.014
4	7.906	0.716	0.746	0.046	0.042	0.038
5	11.876	1.191	0.946	0.080	0.068	0.068
6	15.796	1.198	1.000	0.096	0.083	0.130

Table C.1: Pointwise variogram uncertainty calculated using the three methods presented.

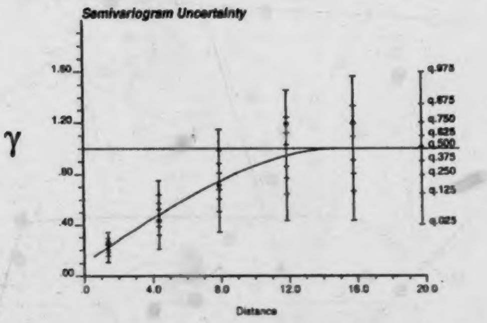


Figure C.3: The experimental variogram, along with the variogram model fitted and the central confidence intervals at 95 %, 75 %, 50 %, and 25 % for each lag (Cluster database).

The variogram, its model and the central confidence intervals at 95 %, 75 %, 50 % and 25 % for each lag are shown in **Figure C.3**.

C.7 Example 2: Red.dat

This database contains samples of a vertical north-south tabular deposit, where thickness and gold, silver, copper, and zinc concentrations were measured. The variogram uncertainty is calculated for thickness and gold content using the Theoretical Approach and both numerical methods. The sample locations are presented in **Figure C.4**. The normal score transformation is performed for each variable. The following isotropic variogram model is fitted to the omnidirectional experimental variogram of thickness:

$$\gamma(\mathbf{h}) = 0.15 + 0.85 \cdot Exp(\frac{h}{250}) \qquad \text{(C.13)}$$

For gold content, the variogram model is:

$$\gamma(\mathbf{h}) = 0.45 + 0.55 \cdot Sph(\frac{h}{250}) \qquad \text{(C.14)}$$

The calculation of confidence intervals was performed for each variable, and the results are shown in **Figure C.5**.

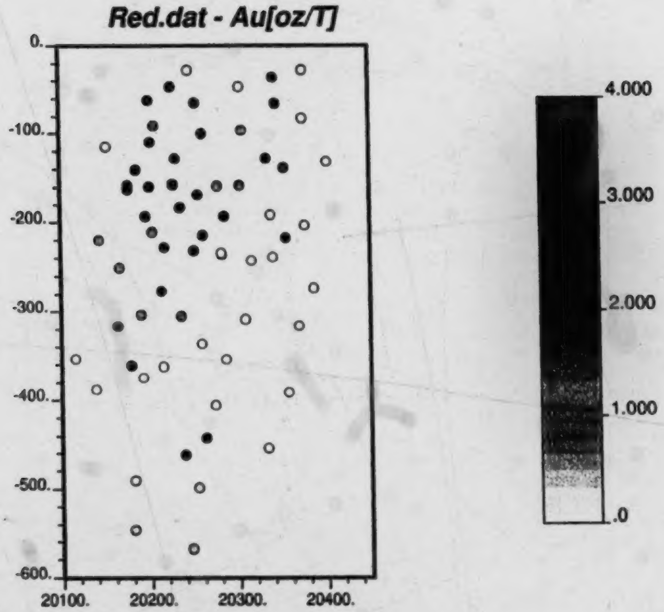

Red.dat - Au[oz/T]
0.
-100.
-200.
-300.
-400.
-500.
-600.
20100.
20200.
20300.
20400.
4.000
3.000
2.000
1.000
.0

Figure C.4: Location map of samples and gold content taken form the Red database.

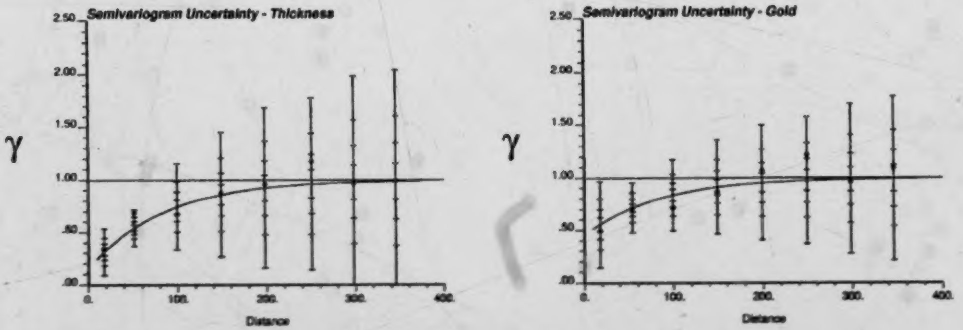


Figure C.5: The experimental variogram, along with the variogram model fitted and the central confidence intervals at 95 %, 75 %, 50 %, and 25 % for each lag (Red database). Left: Variogram for thickness; Right: Variogram for gold content.

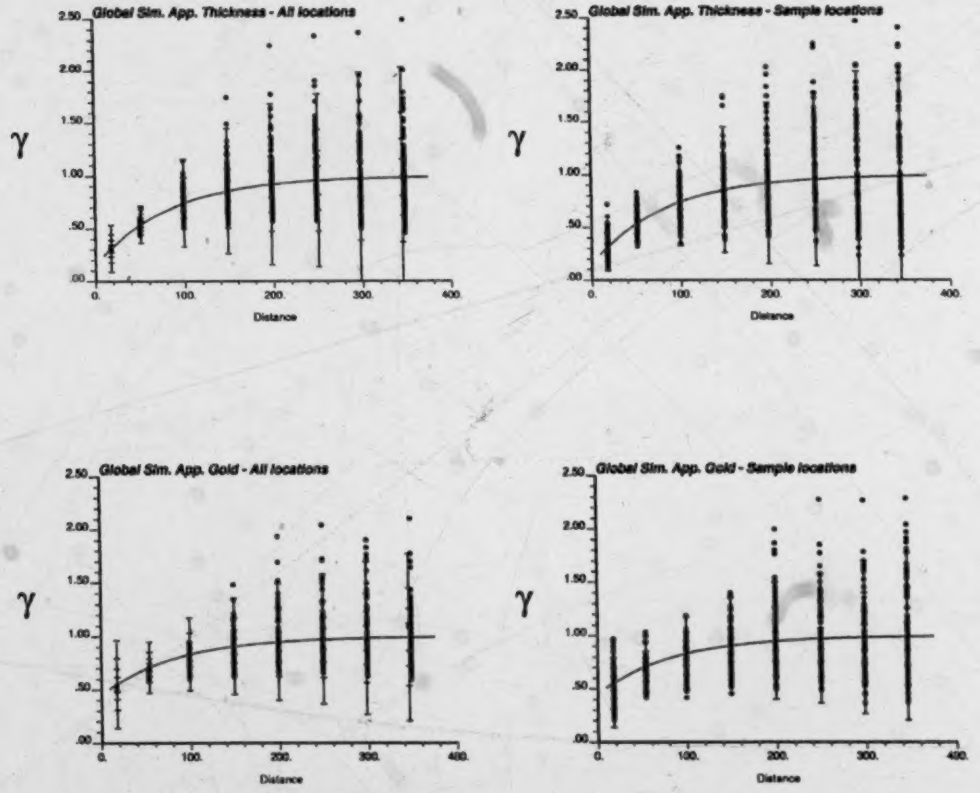


Figure C.6: The experimental variogram values for each lag calculated using (Left) all the simulated data and (Right) only the simulated values at sampling locations (Red database). Top: Thickness; Bottom: Gold.

The Global Simulation Method was used to obtain the entire uncertainty distribution for each lag. 100 non-conditional realizations of a Gaussian random variable were generated using `sgsim`. The simulated values at the sampled locations (obtained from the database `red.dat`) were extracted for each realization. The experimental variogram was calculated using the simulated values at the sampled locations and the same parameters that were used to find the experimental points shown in **Figure C.5**.

The experimental variograms calculated for each realization using the entire simulated field (showing ergodic fluctuations) and those calculated using only the simulated data at the sample locations (now considering the effect of ergodic fluctuations and "sampling fluctuations") are shown in **Figure C.6** for thickness and gold content.

Table C.2 shows the variogram variance for each variable and lag, calculated using the Theoretical Approach, the Local Simulation Method, and the Global Simulation Method. 100 realizations were generated for the numerical methods.

The results obtained from the Theoretical Approach and the Local Simulation Method are similar; however, the Global Simulation Method gives lower variance

<table>
<tr><th colspan="7">Variable: Thickness</th></tr>
<tr><th>Lag</th><th>Lag Distance</th><th>Experimental Variogram</th><th>Fitted Variogram</th><th>Variance Theo. App.</th><th>Var. Local Sim. Meth.</th><th>Var. Global Sim. Meth.</th></tr>
<tr><td>2</td><td>17.497</td><td>0.332</td><td>0.311</td><td>0.013</td><td>0.012</td><td>0.004</td></tr>
<tr><td>3</td><td>51.119</td><td>0.687</td><td>0.540</td><td>0.008</td><td>0.001</td><td>0.007</td></tr>
<tr><td>4</td><td>99.311</td><td>0.669</td><td>0.742</td><td>0.044</td><td>0.041</td><td>0.024</td></tr>
<tr><td>5</td><td>148.627</td><td>0.871</td><td>0.857</td><td>0.092</td><td>0.089</td><td>0.052</td></tr>
<tr><td>6</td><td>197.746</td><td>0.957</td><td>0.921</td><td>0.152</td><td>0.150</td><td>0.085</td></tr>
<tr><td>7</td><td>250.436</td><td>1.178</td><td>0.958</td><td>0.176</td><td>0.177</td><td>0.112</td></tr>
<tr><td>8</td><td>297.843</td><td>0.969</td><td>0.976</td><td>0.264</td><td>0.258</td><td>0.160</td></tr>
<tr><td>9</td><td>345.356</td><td>0.992</td><td>0.986</td><td>0.289</td><td>0.270</td><td>0.193</td></tr>
<tr><th colspan="7">Variable: Gold content</th></tr>
<tr><th>Lag</th><th>Lag Distance</th><th>Experimental Variogram</th><th>Fitted Variogram</th><th>Variance Theo. App.</th><th>Var. Local Sim. Meth.</th><th>Var. Global Sim. Meth.</th></tr>
<tr><td>2</td><td>17.497</td><td>0.493</td><td>0.554</td><td>0.044</td><td>0.041</td><td>0.014</td></tr>
<tr><td>3</td><td>54.099</td><td>0.706</td><td>0.712</td><td>0.015</td><td>0.001</td><td>0.008</td></tr>
<tr><td>4</td><td>99.435</td><td>0.715</td><td>0.833</td><td>0.030</td><td>0.005</td><td>0.015</td></tr>
<tr><td>5</td><td>149.221</td><td>0.865</td><td>0.908</td><td>0.053</td><td>0.043</td><td>0.028</td></tr>
<tr><td>6</td><td>198.912</td><td>1.065</td><td>0.949</td><td>0.078</td><td>0.075</td><td>0.056</td></tr>
<tr><td>7</td><td>249.254</td><td>1.216</td><td>0.972</td><td>0.096</td><td>0.092</td><td>0.066</td></tr>
<tr><td>8</td><td>297.879</td><td>0.961</td><td>0.985</td><td>0.134</td><td>0.140</td><td>0.079</td></tr>
<tr><td>9</td><td>345.618</td><td>1.088</td><td>0.991</td><td>0.160</td><td>0.161</td><td>0.110</td></tr>
</table>

Table C.2: Theoretical approach to calculate the variogram confidence intervals.

for all the lags. The main difficulty of this approach is to ensure correct use of the variogram for all distances when a limited number of nearby samples is used [166]. The variogram calculated for each realization (using all the simulated nodes) was presented in **Figure C.6** (Left). The variability in the variograms calculated using all the nodes in the grid is lower than the expected variability.

Histograms showing the entire uncertainty distribution for the corresponding lags are presented in **Figure C.7**. All the histograms generated through the Global Simulation Method are slightly asymmetric with a tail to the right. This asymmetry was expected since the variogram is non-negative.

C.8 Transferring Pointwise Uncertainty into the Joint Model

Several alternative variogram models could be fitted within the confidence limits generated above. In order to achieve more realistic predictions, we can assume different scenarios within those confidence limits. It is important to note that variogram models fitted using the 97.5 and the 2.5 quantile variogram values for all lags (**Figure C.8**) do not fairly represent extreme cases in the joint uncertainty. The correlation between the lags and the "continuity" of alternative variogram models should be accounted for when fitting models to represent extreme "joint" cases.

Our proposal is to evaluate the consequences of using our first guess (the one used to calculate the pointwise uncertainty), plus two extreme scenarios showing high and low continuity, within the pointwise confidence limits (**Figure C.9**). Simulation can be done using those three scenarios to determine the sensitivity of the results to variogram uncertainty. Notice that we do not just have to modify the parameters (range and sill contribution) of the variogram model, but the type of structure to account for high and low continuity scenarios.

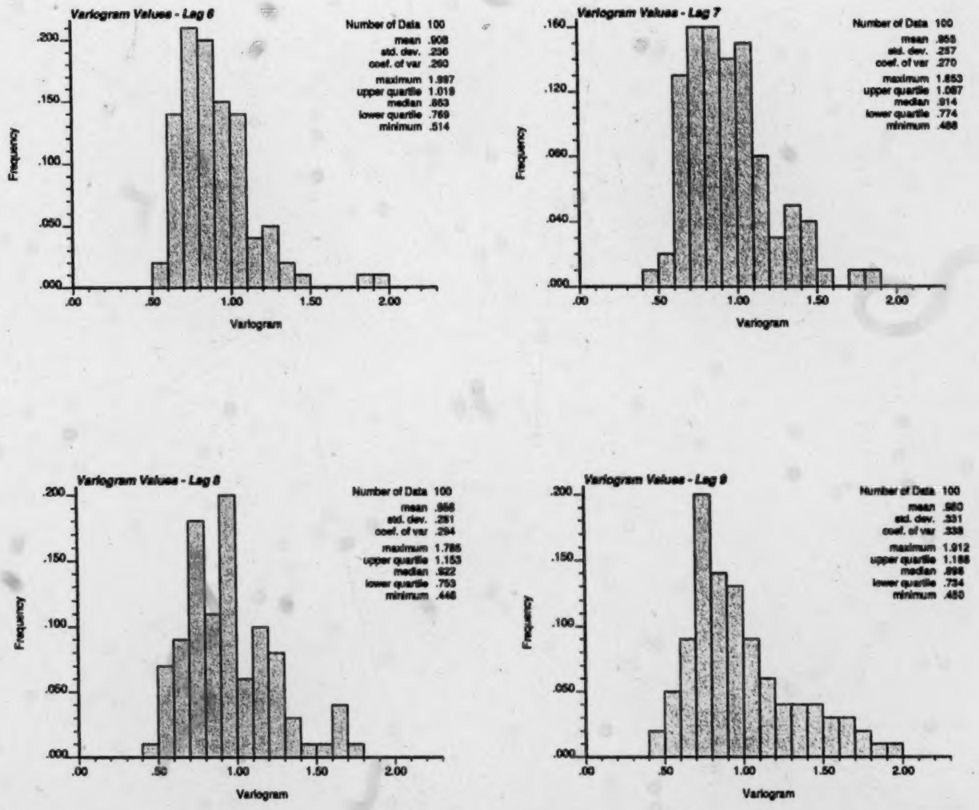


Figure C.7: An example of the uncertainty distribution of the pointwise variogram values: Histograms of variogram values for lags 6, 7, 8, and 9 for Gold (Red database).

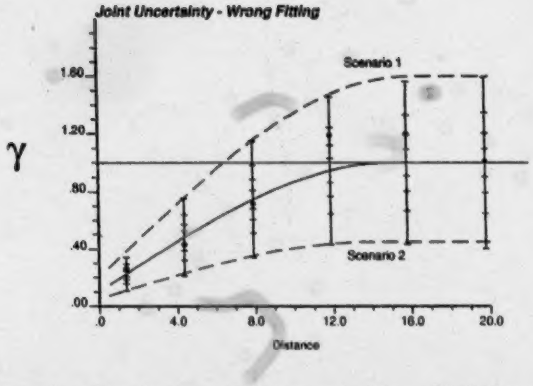


Figure C.8: An example of an incorrect interpretation of joint uncertainty given the pointwise uncertainty. Scenarios 1 and 2 do not represent quantiles 97.5 and 2.5 in the joint model.

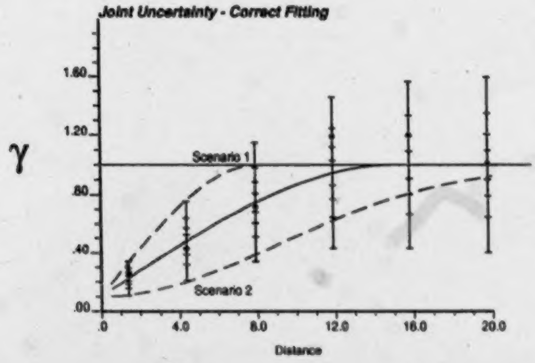


Figure C.9: An example of a correct interpretation of joint uncertainty: Scenarios 1 and 2 represent low and high continuity (extremes of the joint model).

Uncertainty in the variogram sill can be addressed by fitting models with different sill. This uncertainty can be due to uncertainty in the reference statistics.

C.9 Comments

A variogram model is required in all approaches. Ideally, one could determine the uncertainty using the experimental points before fitting a model. The assessment of uncertainty, however, requires a positive definite covariance model (i.e. a non-negative variogram model), therefore a variogram must be fitted before evaluating the uncertainty. This seems circular, however, it is the only way to solve the problem: the authorized model is assumed as the expected value of the variogram at each lag and then the variance is calculated.

The variogram uncertainty can be transferred to subsequent stages of a geostatistical study. The Theoretical Approach and the Local Simulation Method generate the same results. The Global Simulation Method requires more computer time and should give the same result, since the idea is basically the same than the Local Method; however, it is difficult to honor the variogram precisely for large distances and consequently, the variance may be lower. The advantage of the Global Simulation Method is that it estimates the shape of the entire distribution of uncertainty in the variogram for all lags.

Confidence intervals for each experimental variogram value can be determined from the variance assuming normality. This is approximate since the histogram of variogram values obtained for each lag must be non-negative. All methods require multivariate Gaussianity, which could be relaxed with non-Gaussian simulation methods. This has not been explored in this Appendix.

The difference between the point uncertainty and joint uncertainty must be addressed: the procedures presented in this paper allow calculation of the pointwise uncertainty. Within this uncertainty, several variogram models (joint models) can be fitted. The confidence intervals for the joint model will be different since we are interested in finding the uncertainty in the continuity of the variable. Several joint models with different degrees of continuity (e.g. characterized by a Gaussian model the more continuous and by a spherical model the less) should be used in the subsequent stages of the study (simulations, mine planning) to account for the joint uncertainty of the variogram model.

Appendix D

HISIM: Hierarchical Indicator Simulation

D.1 Introduction

The idea of simulating indicators hierarchically in order to avoid order relations and to set a framework suitable to incorporating multiple point statistics was previously proposed. The implementation failed in that indicator variograms could not be reproduced for all thresholds. What initially appeared to be a loss in freedom from one threshold to the next, due to a misinterpretation of the results, was not such. What truly happened was that a virtually random drawing of the nodes were occurring due to the little difference between the probability of informed and uninformed nodes. In this note we explore several ways to fix this problem. A hierarchical implementation of sequential indicator simulation (`SIS`), along with methods that combine the `SIS` paradigm and the hierarchical idea, are also presented. Although some of the techniques here presented gave results quite satisfactory, the problem still remains unsolved from a theoretical point of view.

D.2 The Original Idea

The proposed idea [133] was to simulate one threshold at a time starting at the highest. This can be seen as an eroding algorithm, where all nodes start higher than the highest cutoff, and then they are pushed down based on their probabilities of being below each threshold.

At a given threshold z_k, the conditioning data are coded as indicators:

$$i(\mathbf{u}_\alpha; z_k) = \begin{cases} 1, & \text{if } z(\mathbf{u}_\alpha) \leq z_k \\ 0, & \text{otherwise} \end{cases} \quad k = 1, ..., K$$

where $z(\mathbf{u}_\alpha)$ is the value at the data location $\mathbf{u}_\alpha$.

The idea is to calculate for every node, the probability of it being lower than the current threshold. This is done by simple kriging the indicators. The known mean used to calculate the estimate is the proportion from the global distribution corresponding to the threshold.

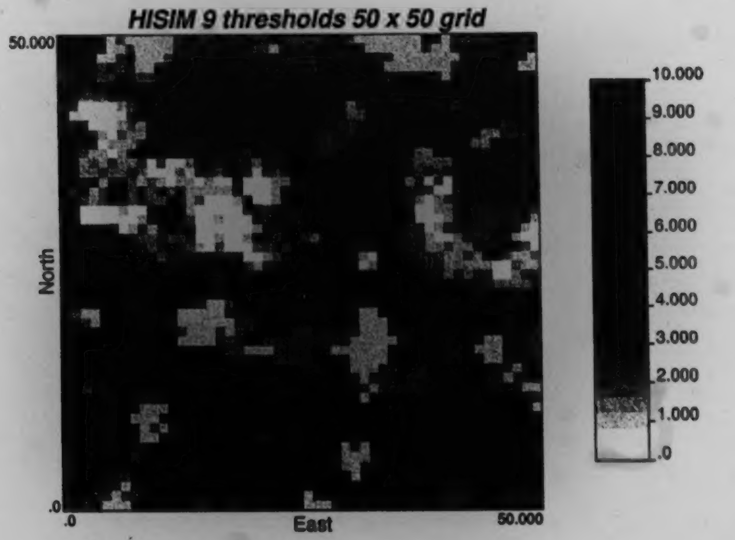


Figure D.1: Map showing the result for the original implementation of `HISIM`. Higher thresholds present high nugget effect.

$$\begin{aligned}[i(\mathbf{u};z_k)]^*_{SK} &= [Prob\{Z(\mathbf{u}) \leq z_k|(n)\}]^*_{SK} \\ &= \sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u};z_k)\cdot i(\mathbf{u}_\alpha;z_k) + [1-\sum_{\alpha=1}^{n}\lambda_\alpha^{SK}(\mathbf{u};z_k)]F(z_k)\end{aligned}$$

where the weights $\lambda_\alpha^{SK}(\mathbf{u};z_k)$ are the unique solution of the simple kriging system.

$$\sum_{\beta=1}^{n} \lambda_\beta^{SK}(\mathbf{u};z_k)\cdot C_I(\mathbf{u}_\beta-\mathbf{u}_\alpha;z_k) = C_I(\mathbf{u}-\mathbf{u}_\alpha;z_k) \quad \alpha=1,...,n$$

Notice that a covariance indicator function $C_I(\mathbf{u}-\mathbf{u}_\alpha;z_k)$ (or, assuming stationarity, $C_I(\mathbf{h};z_k)$), has to be inferred for each threshold.

Once the probabilities are known for every node, a node is chosen based on them, that is, a uniform random number between zero and one is drawn and the nodes are visited in order until the sum of probabilities is higher than the random number multiplied by the total sum of probabilities. In this manner, nodes with higher probability of being below the threshold, i.e. with higher kriging estimates, will have a larger probability of being switched down or eroded.

As in the example shown on **Figures D.1** and **D.2**, the variogram models are not reproduced for higher thresholds, that is for the thresholds that were simulated first in the algorithm.

The initial idea of loosing freedom discussed in [133] from one threshold to the next was therefore a misinterpretation of the results. The proposed correction of using cokriging instead of kriging to calculate the probabilities is also erroneous, since at the first threshold the cokriging estimate is the same than the kriging one, since there is not information at other thresholds than the one being worked on (this is true in an unconditional case).

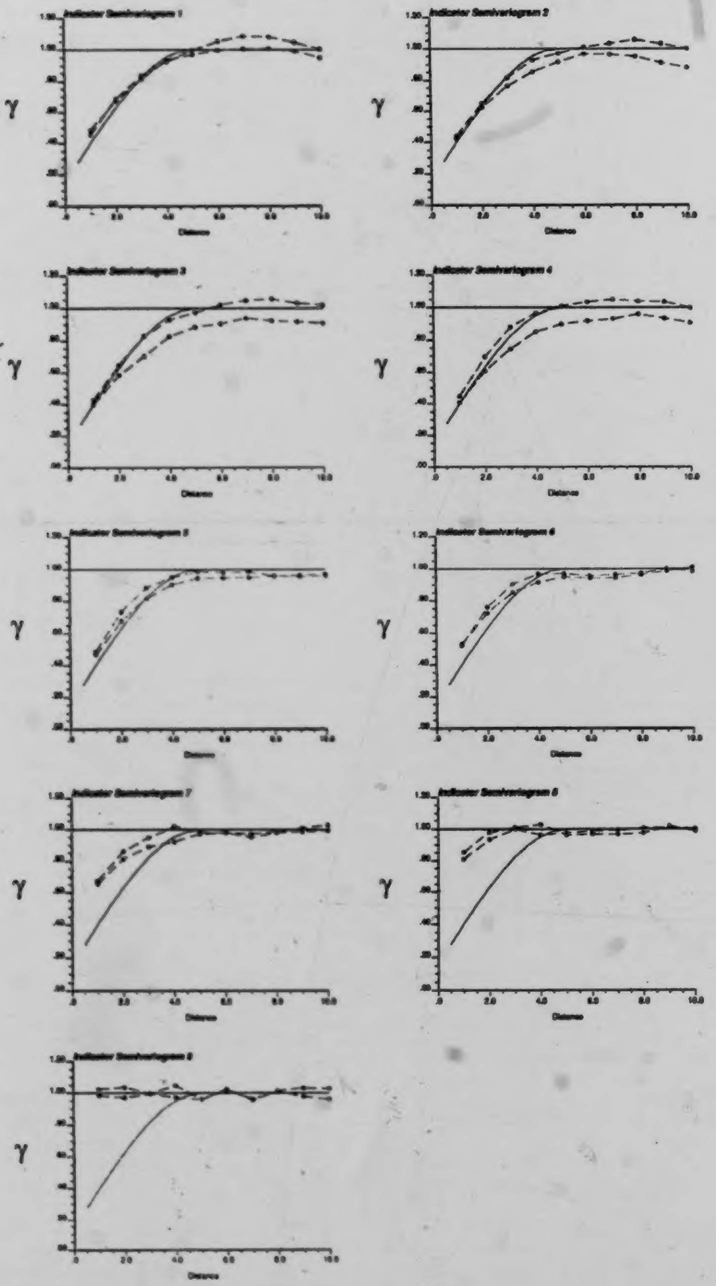


Figure D.2: Variogram reproduction for the original implementation of `HISIM`.

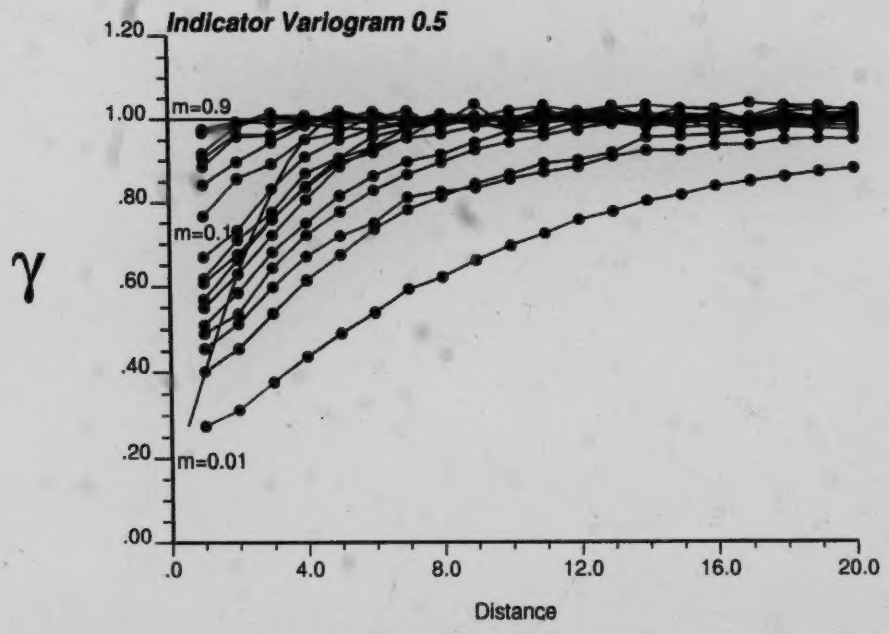


Figure D.3: `HISIM` varying the simple kriging mean for a single threshold case (skmean varying from 0.01 to 0.9).

The nugget effect seen at the highest threshold is due to the extremely little difference between the probability of a node uninformed ($p \in [0.9 - 1.0]$) and a node that has been informed, i.e. switched down ($p = 1.0$). This leads to a virtual random drawing of the nodes. This effect is less severe when a lower threshold is being simulated, since the difference between a node uncorrelated with the conditioning data and the others is larger, so the drawing in not random anymore. Although a very high nugget effect is still present, some correlation can be observed.

D.3 Proposed Approaches

D.3.1 Modifying the Mean in Simple Kriging

The first proposed approach is to modify the mean used when kriging the indicators. A simple example with one threshold at the median is used to test this method. Although intuitively the mean used when simple kriging should be $F(z_1) = 0.5$, where z_1 is the only cutoff, several means were used. The results are not encouraging, since, as seen in **Figure D.3**, a decrease in nugget effect is accompanied by an increase in the correlation range. Therefore, the variogram cannot be reproduced by simply changing the simple kriging mean.

D.3.2 SIS Hierarchical

The idea of eroding an initially high field is now replaced by the hierarchical application of SIS (sequential indicator simulation). The idea is to perform SIS at the

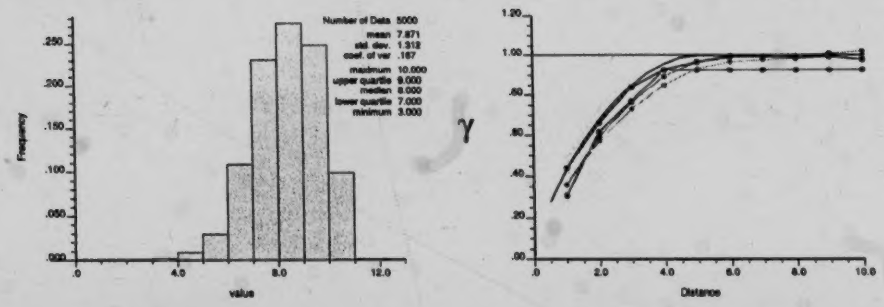


Figure D.4: SIS applied hierarchically. The use of zeros from the higher thresholds biases the conditioning data, generating realizations that do not honor the proportions. The histogram shows that there are no nodes being assigned to the lower thresholds, since they have all already been assigned to higher ones. The standardized variograms using the resulting proportions show that the correlation is preserved.

highest threshold and then use the nodes simulated to be above that threshold as conditioning data for the following thresholds, since it is known that if the node is above a threshold, it is also above all other lower thresholds. This results in realizations that do not honor the proportions required, because of the bias introduced by the conditioning data. They are heavily biased towards zero, since those are the only nodes that can be used as conditioning data when proceeding from the highest threshold down. However, variogram reproduction was reasonable, except for the sill that depends on the proportion of ones and zeros (**Figure D.4**).

This naturally leads to two ideas:

- To use an approach similar to the nested indicators proposed by Dagbert for kriging reserves [26].
- To modify the proportions used as input to obtain the desired ones in the output.

D.3.3 Nested Indicator Simulation

The first solution was implemented with relative success. The steps involved in its implementation are:

1. At the highest threshold, the domain corresponds to all uninformed nodes.
2. An uninformed node is picked in the domain randomly.
3. The simple indicator kriging estimate at the current threshold is calculated given the nearby data and previously simulated nodes.
4. A random number is drawn and a one is assigned to the node if this random number is lower or equal than the simple indicator kriging estimate of the probability at that threshold, and a zero otherwise.

5. Go back to 2 until all nodes in the domain have been visited.
6. If the value is above the threshold, that is a value of zero was assigned in the binary simulation, then eliminate the node of the domain for the next threshold.
7. If the value is below the threshold include it in the domain for the next threshold.
8. Repeat for all thresholds.

In the end, a continuous value can be assigned at every node, since the class to which it belongs is known. The usual interpolation and extrapolation beyond the discrete cumulative distribution function used in `SIS` is required (see for example [39]).

One of the problems of this approach is that correlation between thresholds is not imposed, therefore the result looks patchy, and it is common to find high values beside low values without the appropriate transition in between. This algorithm has been fully developed. Refer to [135] for further details and applications.

D.3.4 Correcting Proportions: Markov and Empirical Approaches

The second proposed solution implies accounting for the bias generated by the conditioning data. The question is: How much do we have to change the input proportion to obtain the required proportions?

After several attempts, a correction factor for the proportion used as a mean was applied. This implies a non-linear additive correction to the estimated probabilities. Consider the original estimate, using P_{Theo}, and the new estimate using P_{Corr}.

$$\begin{aligned}
[i(\mathbf{u})]^*_{Theo} &= \textstyle\sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u}) \cdot i(\mathbf{u}_\alpha) + [1 - \sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u})] \cdot P_{Theo} \\
[i(\mathbf{u})]^*_{Corr} &= \textstyle\sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u}) \cdot i(\mathbf{u}_\alpha) + [1 - \sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u})] \cdot P_{Corr}
\end{aligned}$$

The difference in the estimate is:

$$\Delta = [1 - \sum_{\alpha=1}^{n} \lambda_\alpha^{SK}(\mathbf{u})] \cdot (P_{Corr} - P_{Theo})$$

Next to a data location, this factor vanishes, since, the sum of the kriging weights approaches one. On the other hand, far from data, this factor tends to its maximum, $P_{Corr} - P_{Theo}$.

Notice also that the same type of correction would be possible using a cokriging approach. The correlation between indicators at different thresholds does not need to be input. It can be calculated, given the proportions of ones at the current threshold p_2, and the proportion of ones at the previous (higher) threshold p_1:

$$\rho = \sqrt{\frac{p_2 \cdot (1 - p_1)}{p_1 \cdot (1 - p_2)}}$$

We experimented also with this approach. Results showed that the proportions were not reproduced either. Variograms showed a small increase on the nugget

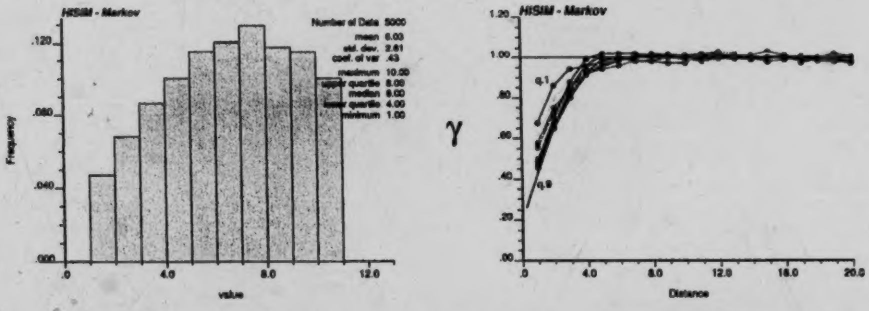


Figure D.5: Hierarchical application of SIS using a Markov assumption for collocated cokriging of the indicators using the value at the previous higher threshold. The histogram is not reproduced (uniform distribution) and some increase in the nugget effect can be seen for lower thresholds.

effect for lower thresholds, i.e. the last ones being simulated. However, the range of correlation was preserved (**Figure D.5**).

An empirical correction factor for the simple kriging mean was found that "updates" the mean for every threshold. It is the ratio of the average probability expected for each node at the current threshold, over the average probability calculated considering the conditioning data:

$$f = \frac{p_2}{\frac{\sum_{i=1}^{nx} i^*_{SK}}{nx}}$$

where p_2 is the proportion at the current threshold, nx is the total number of nodes, and i^*_{SK} are the simple kriging estimates of the probabilities of being below the threshold.

The simple kriging mean is then multiplied by this factor every time a new threshold is being simulated.

The results show a good reproduction of the histogram, but an increase in correlation for lower thresholds, along with a decrease in nugget effect as the simulation proceeds (**Figure D.6**).

D.3.5 Median Hierarchical Indicator Simulation

One last idea proposed is the use of SIS to simulate at the median, and then proceed up and down using the original hierarchical idea, that is, eroding in both directions, keeping the nodes set below the median when going to higher thresholds, or the nodes set above the median when going to lower thresholds. This is illustrated in **Figure D.7**.

This algorithm proceeds as follows:

- Simulate by SIS (or any other binary simulation method, such as truncated Gaussian simulation) the median threshold. Every node is assigned a one or a zero, depending if they are below or above the median value, respectively.

- For thresholds below the median:

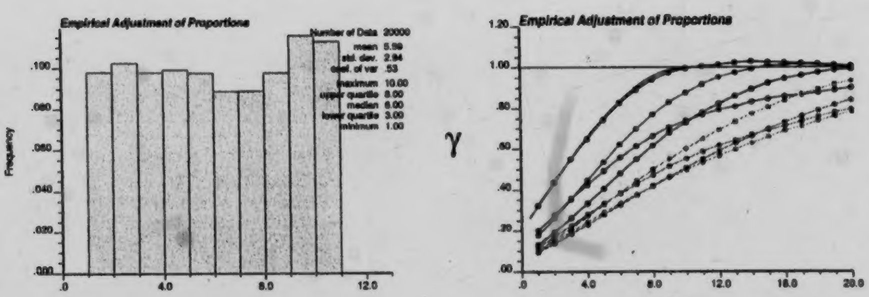


Figure D.6: Empirical adjustment of the proportion to apply SIS hierarchically. Histogram reproduction is good, variograms show an increase in correlation and reduction in nugget effect.

	Node																			
Threshold	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0.9	1	1				1		1	1	1	1				1	1			1	
0.7	1	1				1		1	1	1	1				1	1			1	
0.5	1	1	0	0	0	1	0	1	1	1	1	0	0	0	1	1	0	0	1	0
0.3			0	0	0		0					0	0	0			0	0		0
0.1			0	0	0		0					0	0	0			0	0		0

Figure D.7: Illustration of median hierarchical indicator simulation. The nodes with a value higher than the median are used as conditioning data for lower thresholds, and the nodes with values below the median are used as conditioning data for all thresholds higher than the median.

  - Use the nodes set above the median, that is those coded with a zero, as conditioning data.
  - Calculate the simple indicator kriging estimates at every location.
  - Select one location by Monte Carlo drawing, using the probabilities previously calculated by simple indicator kriging.
  - Repeat until the right proportion of nodes has been set below the current threshold.
  - Set all the nodes that have not been switched as zero. Their values are between the current cutoff and the higher threshold.
  - Use all the nodes with zero values (the ones that have just been coded and those that were coded in a previous threshold simulation) as conditioning data for the next threshold.
  - Repeat until the lowest threshold has been simulated.

- For nodes above the median:
  - Code the data above the median as ones and the nodes with values below the median as zeros.
  - Proceed as with the thresholds below the median, but working with the probability of being above the cutoff, instead of below it.

The algorithm is symmetric with respect to the median. Results showed good reproduction of the histogram: the number of nodes above and below the median presents ergodic fluctuations from `SIS` or the algorithm used to generate this binary simulation. The proportions for other thresholds is guaranteed by construction since the number of nodes to switch is defined by the proportions. Variogram reproduction at the median is also obtained depending on the algorithm used to generate the initial binary simulation. At other thresholds, variogram reproduction is obtained just as in the original case, but here, the problem of having a small difference between the probability calculated by simple kriging and the probability for nodes away from data is large, so the drawing of the nodes to be switched is not random (**Figure D.8**).

A first example is shown in **Figure D.9**. Twenty realizations of a one dimensional array of 3000 nodes with intrinsically correlated indicators, that is the so called mosaic random function model, was tested. Nine thresholds and a spherical variogram with a range of 10 units and a 10% of nugget effect was used. The results are encouraging. Variogram reproduction is good, although a slight increase in correlation can be seen for indicators far away from the median.

A second example with a multivariate Gaussian variable is also presented (**Figure D.10**). In this case variogram reproduction is poor at thresholds other than the median. However, the range of correlation is preserved. Again histogram is reproduced by construction.

Finally, a non-Gaussian variable was used (**Figure D.11**). The results are a mixture of the previous two examples. Good reproduction of the indicator variograms at some thresholds and poor reproduction at others.

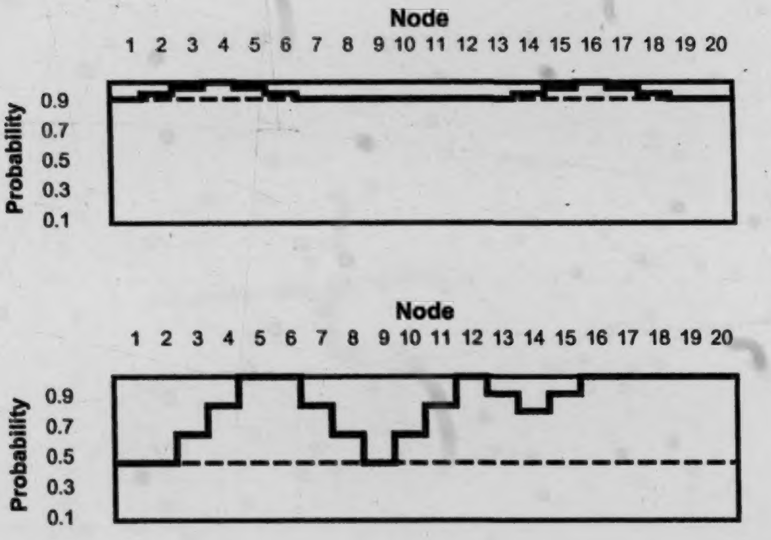


Figure D.8: Illustration of the case when drawing nodes by Monte Carlo simulation is virtually random (top) and when the drawing is effective and accounts for the now larger differences in probabilities (bottom).

D.4 Conclusions

Simulating one threshold at a time is an appealing idea, since this avoids order relation deviations, and permits a useful framework for incorporating multiple-point statistics.

The original idea failed in that correlation could not be reproduced for high thresholds. However, correlation was recovered as the algorithm proceeded to the lower thresholds. The use of another technique such as SIS for locking the realization at a given threshold was explored, however, variogram reproduction was never achieved in a completely satisfactory way. Apparently, the biased conditioning generates unavoidable bias in the covariance reproduction. This problem is difficult to tackle, since we proceed sequentially, and this generates a constant change in the magnitude of the bias. The idea of correcting while simulating could be a possible way to fix this problem.

Among all the techniques explored, the nested approach seems reasonable, because it rests in the well known indicator approach. Research could focus on correcting for the increase in nugget effect generated by not accounting for the zeros from the higher thresholds. The result would be different than the one obtained through SIS, since the nested approach would generate a map that truly resembles a mosaic, in the sense that patches of different classes would be randomly distributed in the field.

As a final comment, the incorporation of multiple-point statistics could be approached separately from this hierarchical algorithm. Runs could be drawn directly into a field without even considering the two-point statistics.

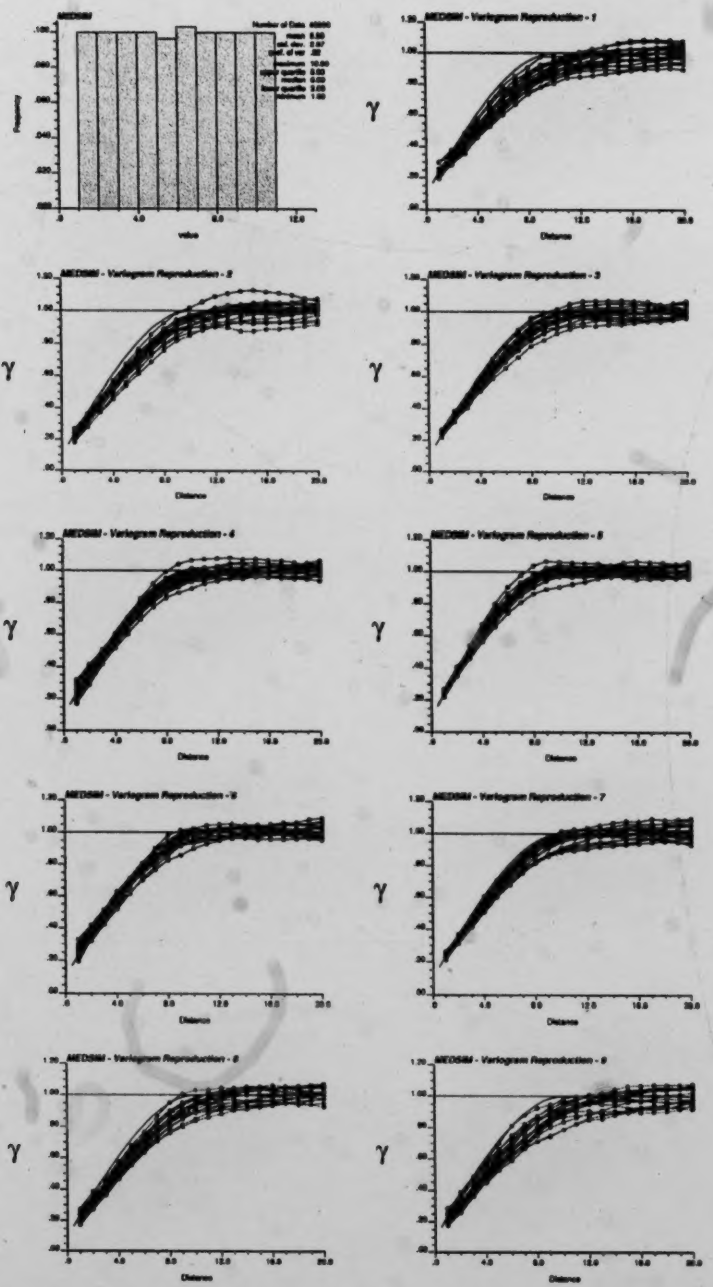


Figure D.9: Application of median hierarchical indicator simulation for an intrinsically correlated variable or mosaic model.

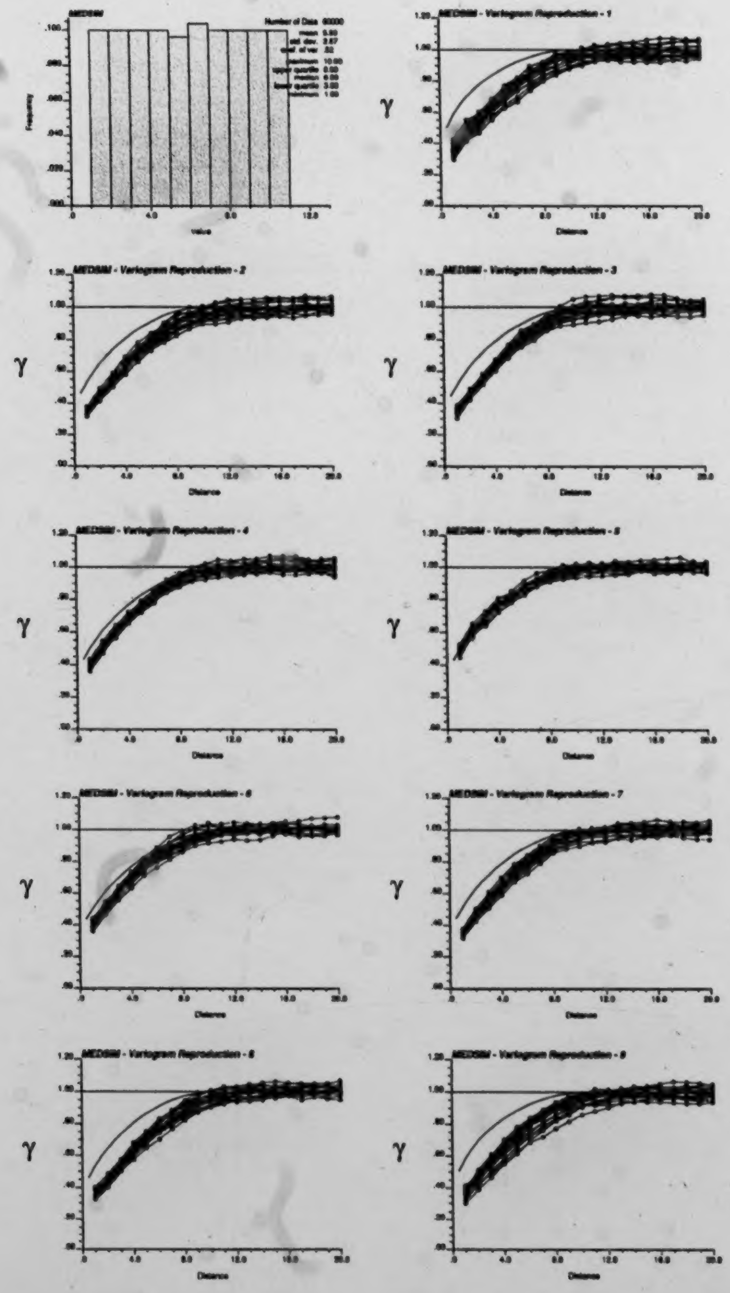


Figure D.10: Application of median hierarchical indicator simulation for a multi-Gaussian variable.

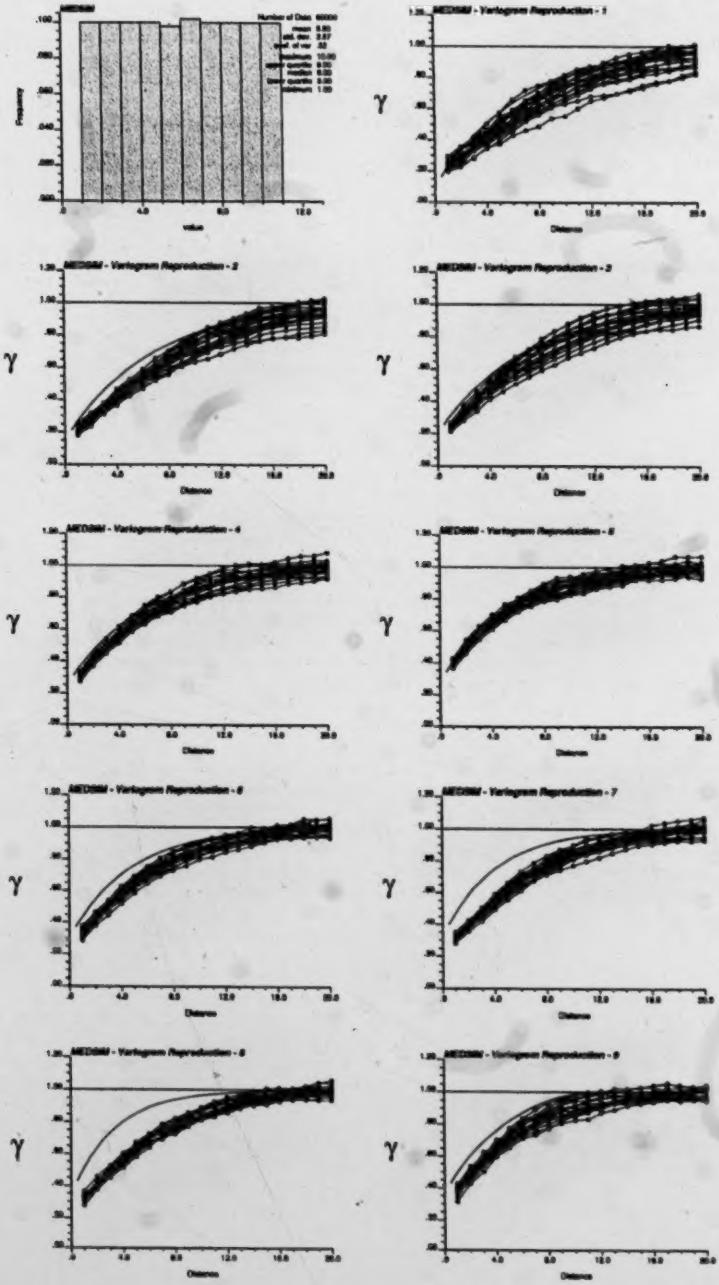


Figure D.11: Application of median hierarchical indicator simulation for a non-Gaussian variable.